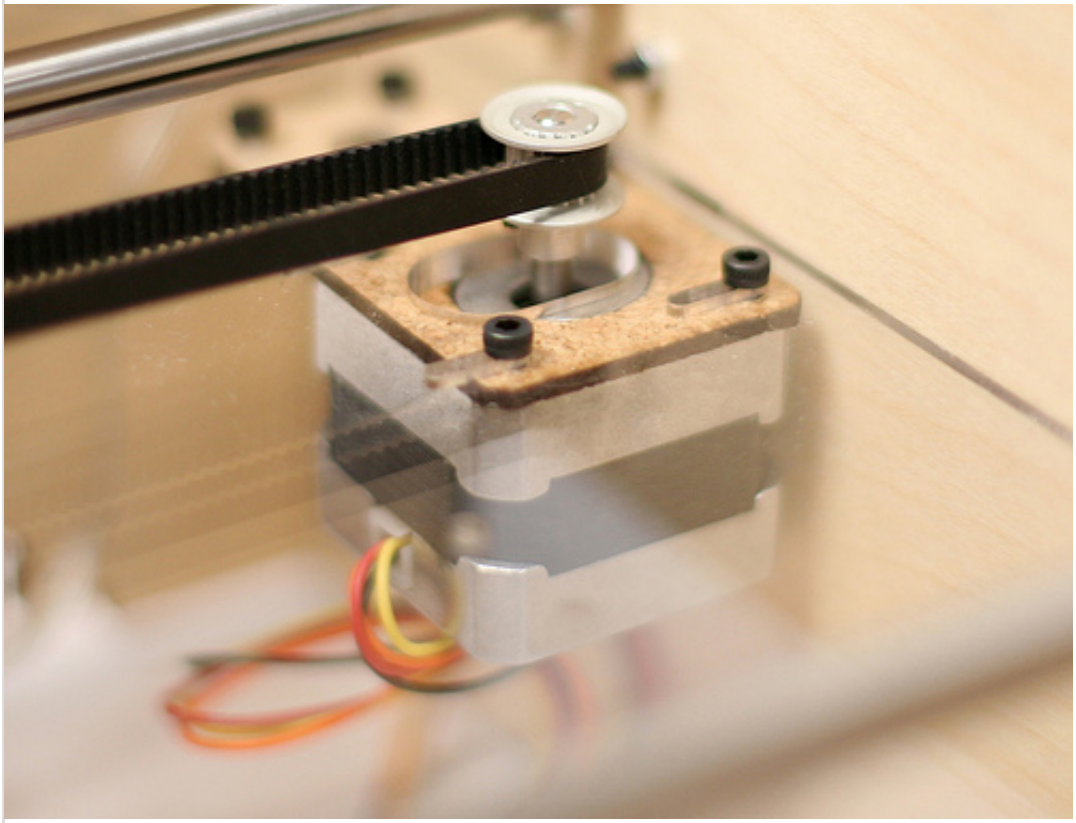
Proper Belt Tensioning

Tensioning Mechanism

The tensioning mechanism on the Thingomatic is the motor mount. Instead of holes, there are slots where the motor is mounted. This allows the motor to move in a direction parallel to the axis. To loosen or tighten the belt, loosen the bolts on the motor and slide it towards or away from the stage. When you are satisfied, tighten the motor bolts back down.

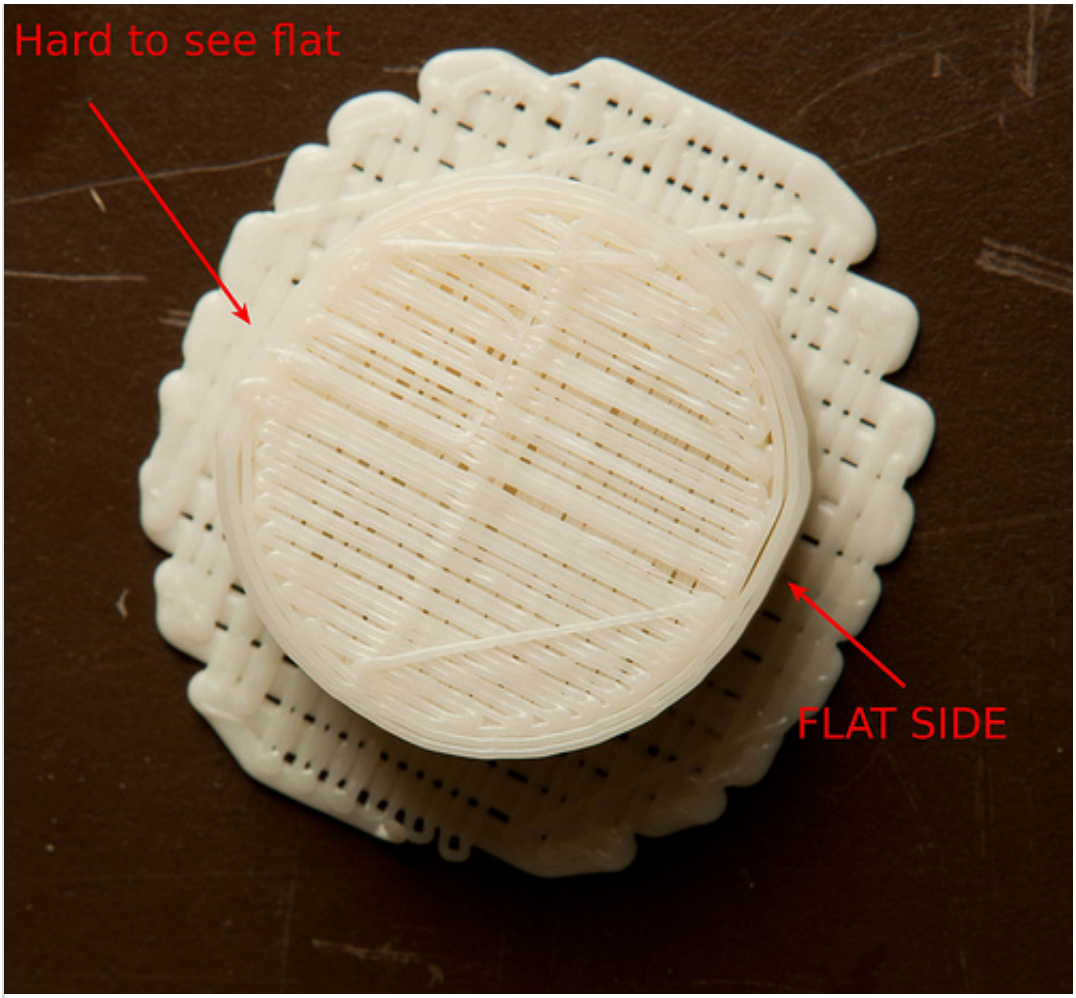
This may be a bit tricky on the X motor, and you will need to take the top plate off the Y carriage in order to get access to the bolts.



Tensioning belts

In general, belts should be tight enough to minimize slack, but not so tight that they start placing a lot of stress on the motor shaft or pulleys. Once a belt is on, turn the motor pulley with your fingers to gauge if there's too much resistance. If the belt makes an audible noise when you 'pluck' it with your fingers, it is too tight. The Thingomatic operation should be nearly silent. If your motor makes a humming or buzzing noise, your belts may be too tight.

To test if your belt is too loose, rotate the motor pulley back and forth by hand. If the stage moves back and forth in sync, then you are okay. If it lags behind, the belt is too loose. A more conclusive test is to print a cylinder from Thingiverse on your Thingomatic. If any of the sides are flattened, then the belt on the corresponding axis is too loose.

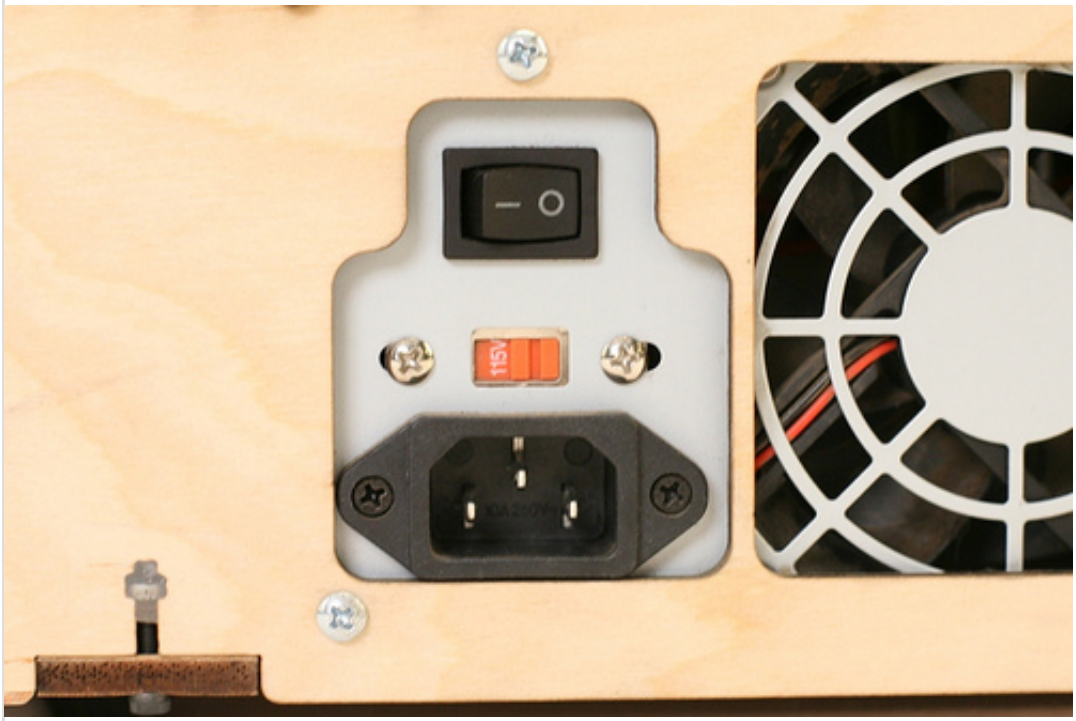

Hard to see flat
FLAT SIDE

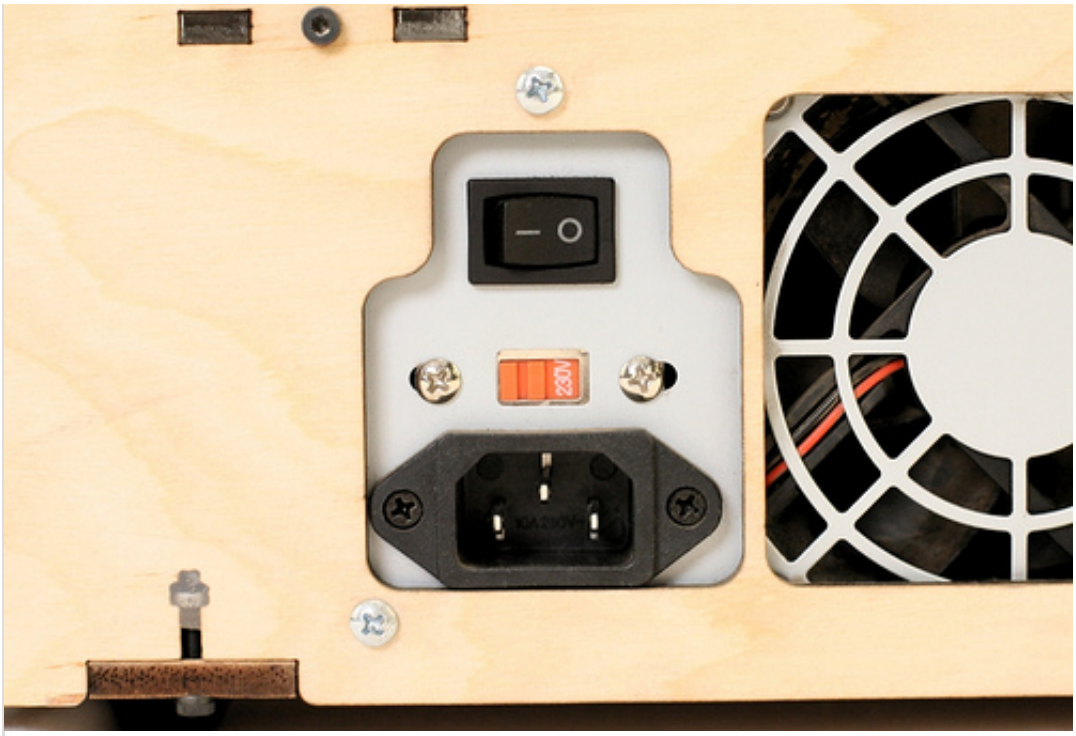
Testing Your Bot

Connect to your bot

Double-check your voltage

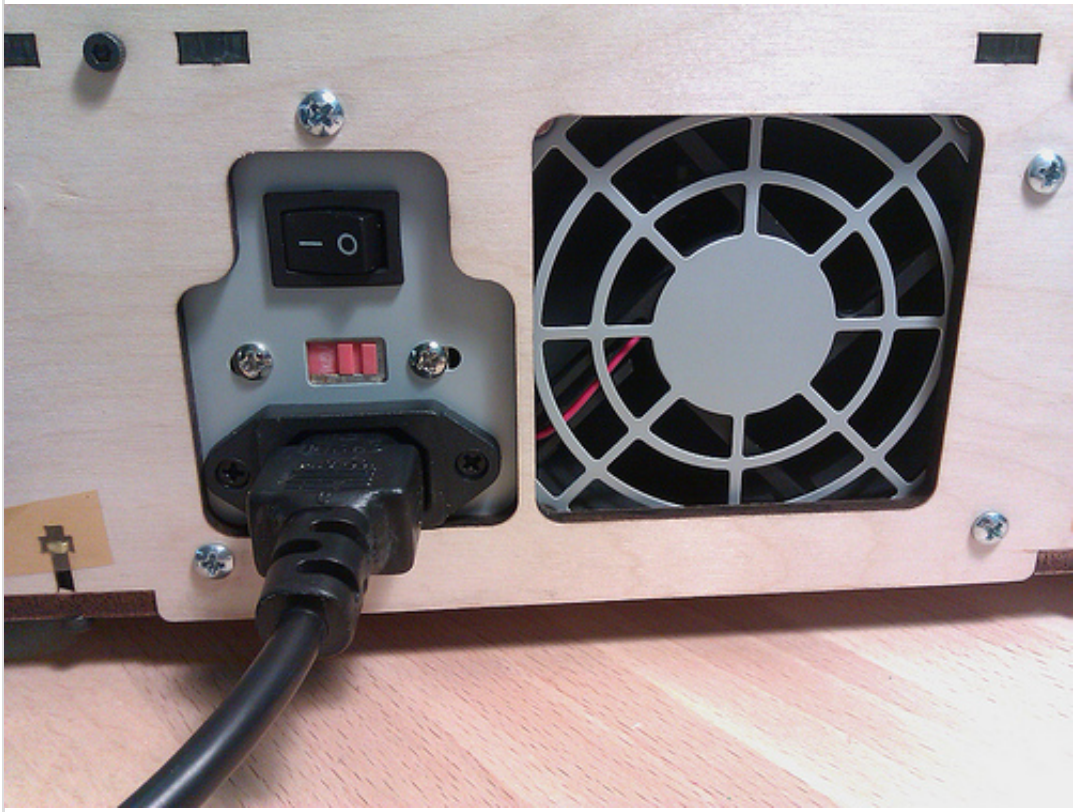
This is also a good time to double-check that the red switch on the back of your power supply that selects your local voltage level is in the correct position. The switch should be pushed to the *right*, with the text "115V" visible, in areas with 120V power mains (USA and Canada). The switch should be set to the *left*, with the text "230V" visible, in areas with 230V power mains (EU, China). Make sure you don't have the power supply set to "115V" when plugging it into a 230V outlet!





Plug in the power and turn it on

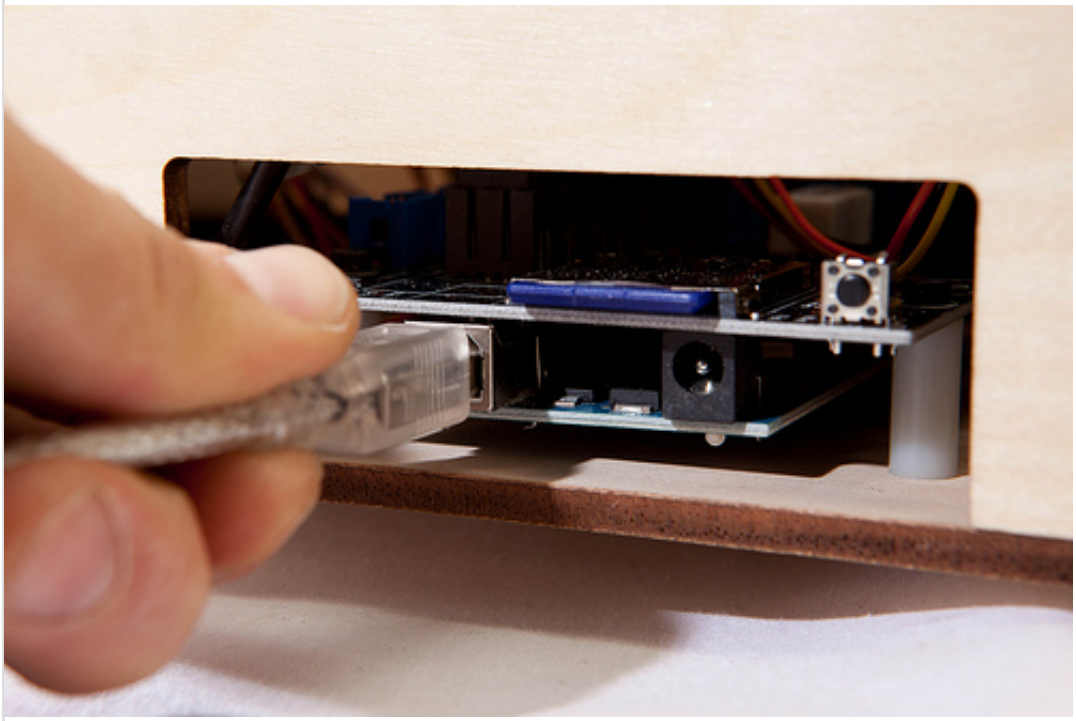
Find the black power cord provided in the kit. Plug one end into the power supply on the left side of your machine and the other end into a wall socket. Flip the switch on the power supply to ON. The fan on the power supply should start up.



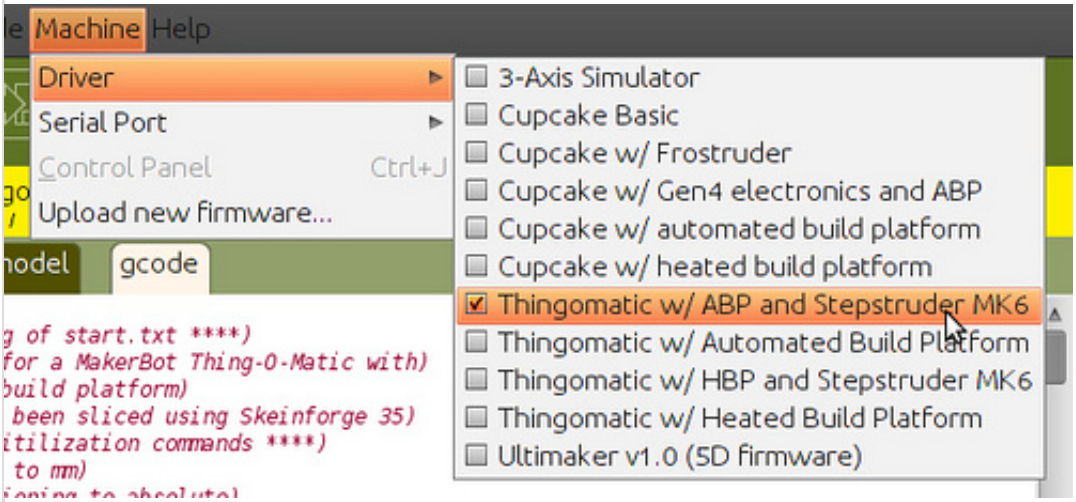
Plug In USB and Connect

Take the included USB A to B cable and plug one end into the Arduino MEGA on the right side

of your bot. Plug the other end into your computer. Next, open up the ReplicatorG program that you installed in the very first step.



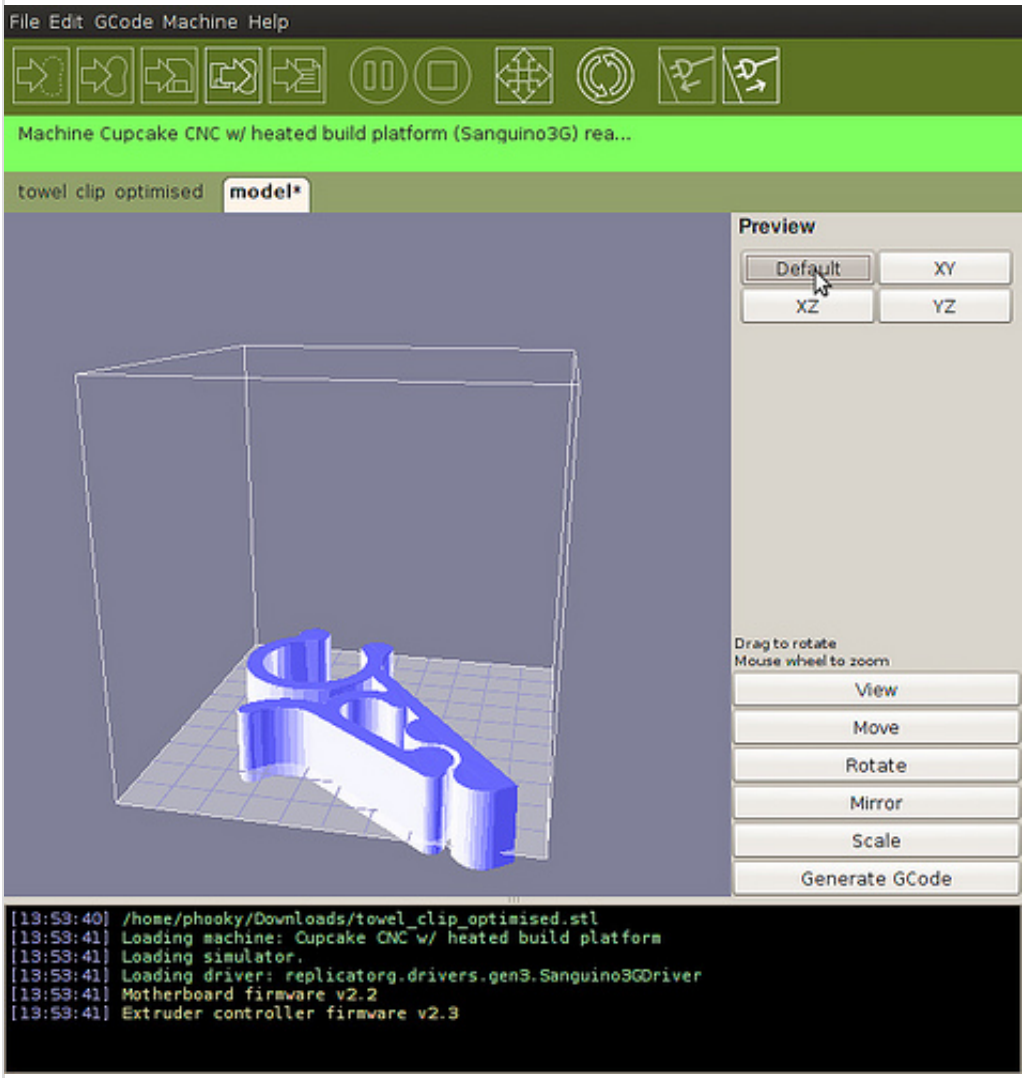
First, select the appropriate Machine -> Driver -> Thingomatic driver for your build platform. If you've just built a MK6-equipped bot, make sure that you select on of the options with "MK6" in the title — otherwise your stepper motor won't move correctly.



Now, you'll need to find the correct serial port that your machine has registered as. You can see a list of connected serial ports in the Machine -> Serial Port submenu. If your computer doesn't have a built-in serial port, you may only see one entry; if you're using a desktop with built-in serial ports, you'll probably see several entries. The way a Thing-O-Matic appears in this menu depends on the platform you're running on. On Windows machines, it may appear as "COM3:". On Linux machines it often appears as "/dev/ttyUSB0". Macs will see multiple entries for each serial port, with a unique ID associated with each physical connection. The entry you'll want to choose on a Mac is one of the form "/dev/tty.usbserial-XXX", where XXX is a unique identifier.

After you select a serial port, ReplicatorG will try to connect. (If you've got the "Automatically connect on startup" option unchecked in the preferences panel, you'll may need to press the

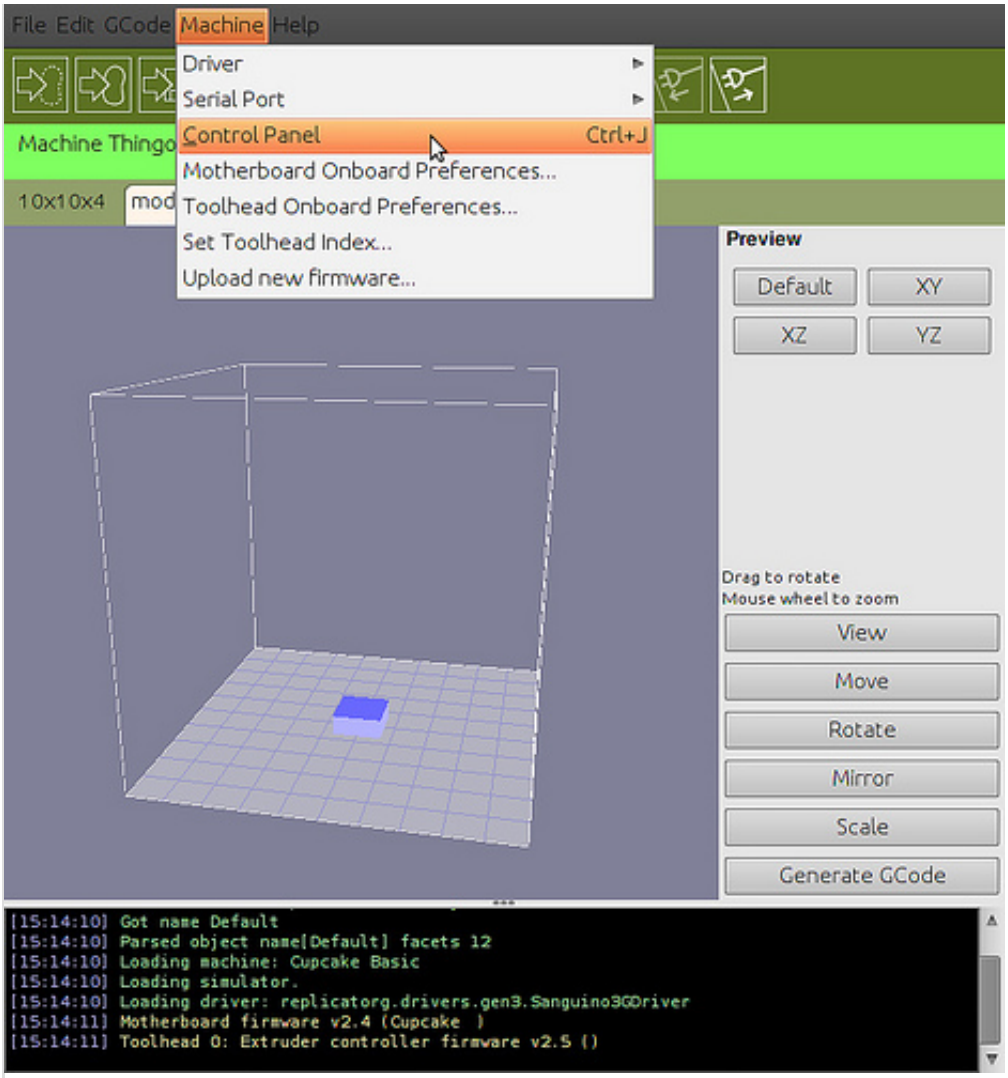
'Connect' icon on the right side of the button bar to tell ReplicatorG to connect.) The status bar should turn green, as shown. **The console window below should list both a Motherboard and Firmware version.**



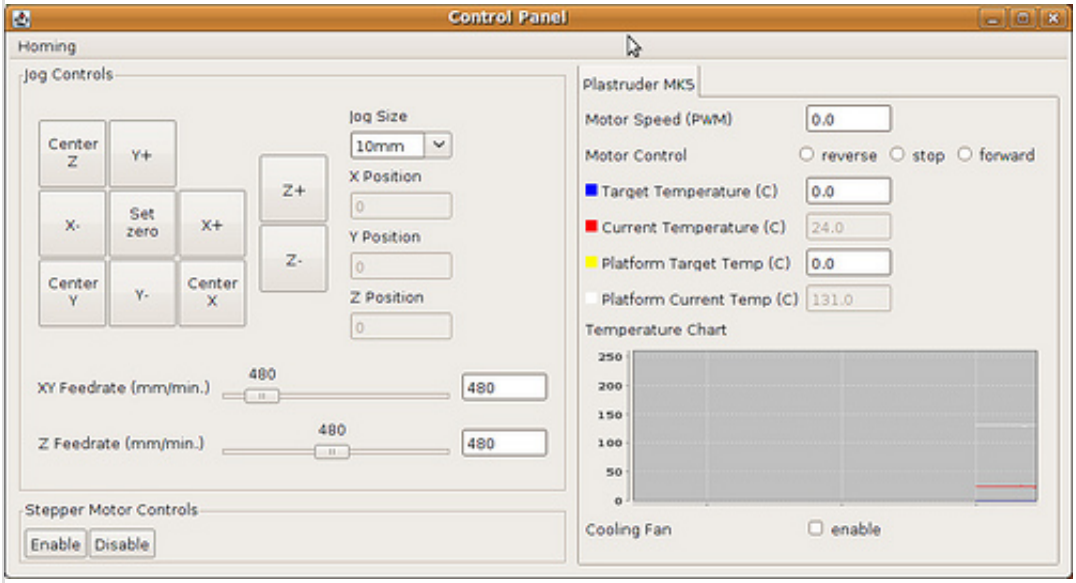
If the console window reports that the motherboard can not connect to the toolhead, make sure your patch cable is connected between the motherboard and extruder controller, that the extruder controller is plugged into the power supply, and that the power on the Thing-O-Matic is switched on.

Step 3: Open your Control Panel

Open the control panel by selecting the Machine -> Control Panel.



You should be presented with this window.

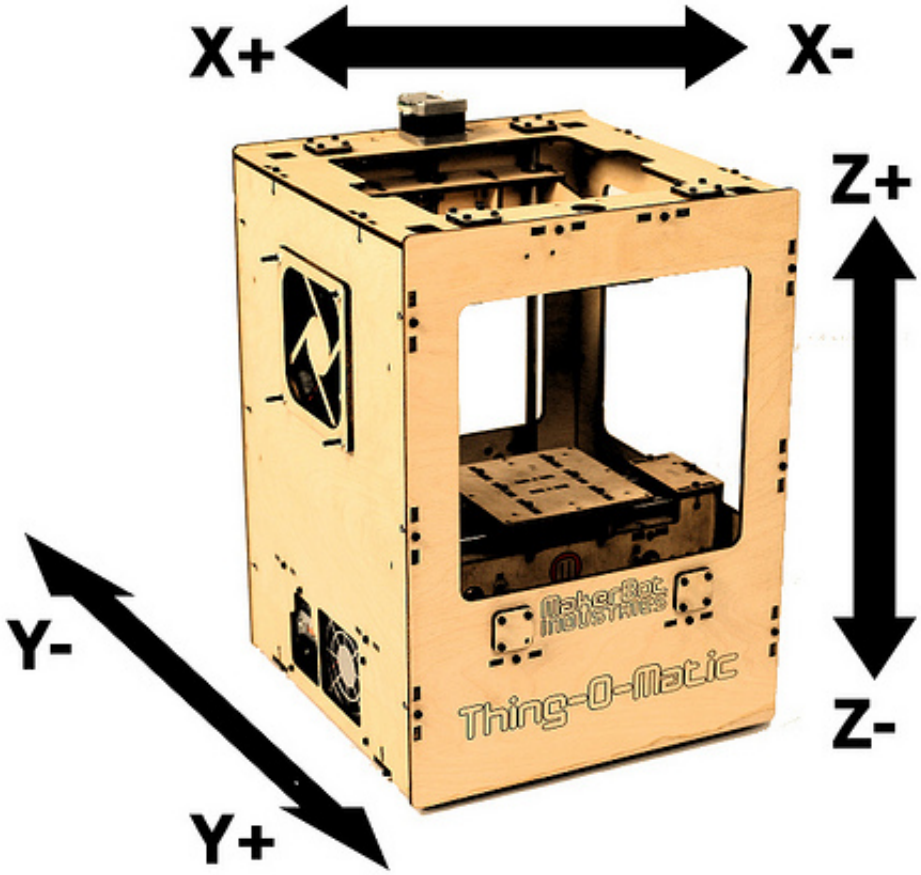


Testing Your Axes

The first step to testing your bot is to verify that all of your stepper motors are working. These are the motors that move your bot around, and if they don't move, then you don't print. First,

there is some information you need to know about your XYZ axes configuration.

Get to know your axes



The Thing-O-Matic moves along three axes: the X axis, the Y axis, and the Z axis. The X axis is the one that runs from left to right; the Y axis is the one that runs towards and away from you. (Think of these as being like the X and Y axis of a graph, seen from above.) The Z axis goes up and down.

Spend a moment to familiarize yourself with the directions of the axes. It will save you a lot of frustration later!

XY Axes and Relative Motion

Look closely at the picture above. On first glance, the X and Y axes may look backwards to you. Moving the X axis in the negative direction means moving it to the right. The reason for this is what we call **Relative Motion**. To be specific, we're talking about the motion of the extruder nozzle relative to the build surface.

Imagine for a second you are a tiny little person that is standing on the build surface. You are facing the back of the machine. Suddenly the build surface you are on moves in the X– direction (to your right). You look up at the extruder nozzle and see that it is (from your perspective) moving to the left. The relative motion of the nozzle to the build surface that has been achieved is a movement in the negative direction.

When you look at the control panel in ReplicatorG, you'll see the "X+" button is on the right. When you click it to move the platform, though, you'll see the platform move to the *left*. This is expected! The controls in ReplicatorG indicate the *relative motion of the extruder*, not the *motion of the platform*.

First Axis Movement

WARNING: MAKE SURE YOUR PLATFORM IS CENTERED IN THE MACHINE BEFORE YOU BEGIN. You don't want to jam your axes into the sides on your first run! The XY axes can be moved by hand, and the Z axis can be raised/lowered by rotating the drive screw.

Do not force the axis movement! If you clicked one of the buttons, the stepper driver for that axis remains active in 'holding mode'. This means it will fight you to keep the axis in the place it currently is. To turn the motor off, click the 'Disable Steppers' button.

Now it's time for your bot to take its first baby steps. On the left side of the control panel, there are buttons labeled X+, X–, Y+, Y–. These buttons when pressed will tell your Thing-O-Matic to move the extruder head in these directions. Remember: this movement is movement of the extruder head relative to the build surface!

Test that your MakerBot moves each axis in both directions. The table below tells you what direction each button should move each axis:

Button	Movement Seen From Front
X+	platform moves to your LEFT
X–	platform moves to your RIGHT
Y+	platform moves TOWARDS you
Y–	platform moves AWAY from you
Z+	nozzle moves UP
Z–	nozzle moves DOWN

If an axis is moving the wrong way there are two ways to fix this:

1. Shut down ReplicatorG. Turn off the machine, unplug it from both the wall and the USB connection, open the electronics bay, and rotate the stepper connector 180 degrees. This will invert it.
2. Or, you can invert it in software by going to Machine -> Onboard Preferences and clicking the appropriate button to invert that axis.

Secondly, you are looking to make sure that each axis moves nice and smoothly. It should make a humming noise, but any grinding noise or a lack of movement means something is wrong.

If you're having trouble with your stepper motion, take a look at the Thing-O-Matic troubleshooting page.

Crank up the speed

There are sliders underneath the movement buttons labeled XY Feedrate and Z feedrate. These set the speed of the axis movements. You can either gradually increase these or drag them all the way over. Once you do, repeat the back and forth movements from before. The axis should continue to move nicely, except this time at maximum speed.

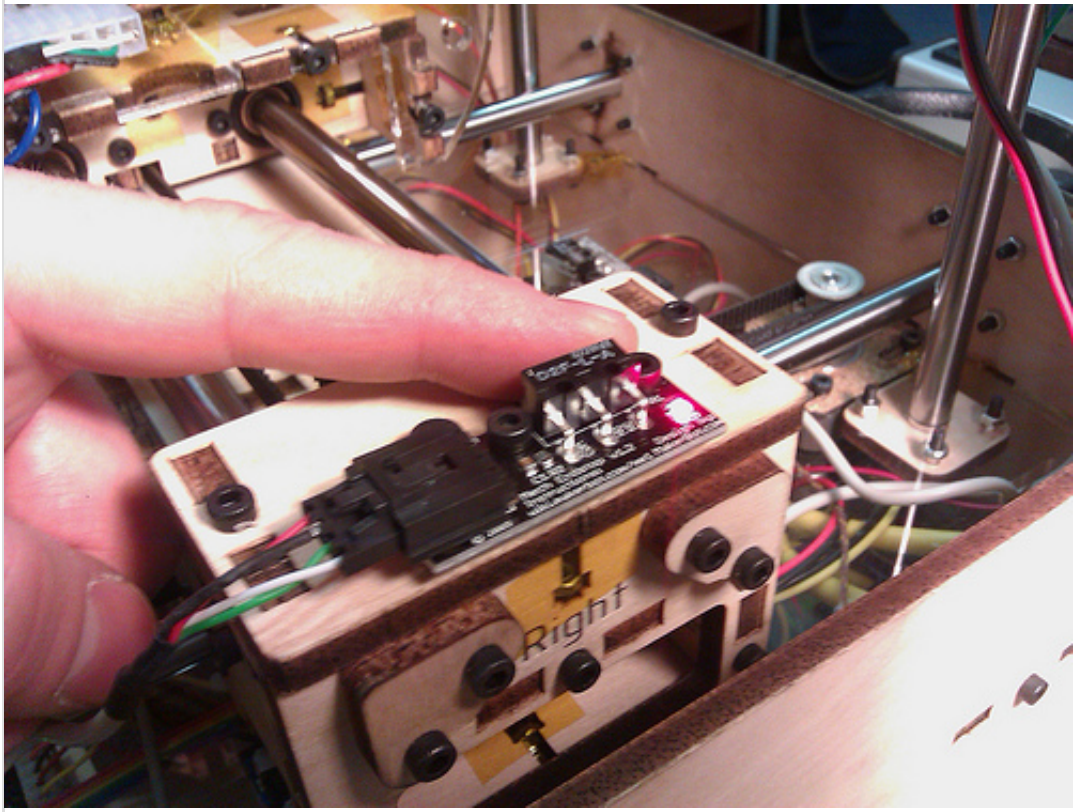
If you're having trouble with your stepper motion, take a look at the Thing-O-Matic troubleshooting page.

Test Your Endstops

There are 3 endstops provided in your Thing-O-Matic kit. They are used to detect when your Thingomatic has reached a certain point, which allows your machine to automatically determine where it is from any position and automatically start printing in the same spot, even if something went wrong.

Step 1: Trigger Each Endstop By Hand

The endstop is a simple mechanical switch. Press it down and you'll hear a nice, satisfying click. If everything works right, you'll also see the little LED on the endstop turn on as well.



If the LED does not turn on, you probably have a wiring problem. Check the electronics installation page and make sure you hooked it up right.

Step 2: Trigger Each Endstop By Machine - Manually

Click the 'Disable Steppers' button in the control panel. We'll be moving the axes by hand first to make sure they all can actually reach their endstops.

One by one, move each axis to its appropriate endstop. Do not touch the endstop itself, but rather move the axis so that it triggers the endstop. The endstop should light up like it did above.

If any axis is unable to trigger its endstop, make sure there are no obstacles and try it again. If there are obstacles, make sure they are permanently out of the way, otherwise it could cause problems during printing.

Step 3: Test Each Endstop For Correct Wiring

Move each axis to the opposite side of its movement from the endstop. Next, turn the speed for each axis down to about 500 for XY and 100 for Z. You will be moving each axes towards its endstops, and triggering the endstops with your finger (or some other object).

Test each axis in sequence by holding down the appropriate endstop and pressing the corresponding button for that axis (X–, Y–, and Z+). The axis should not move while the endstop is triggered.

If this does not work, you most likely have the endstops plugged into the wrong headers on the motherboard. This is a simple mistake to fix, so open up the electronics bay and make sure they are plugged into the right headers.

Step 4: Home Each Axis

This is the final test: can the machine automatically move itself to the home position for each axis? At the top of the Control Panel, there is a 'Homing' menu. Under this menu there are a variety of options that will automatically 'Home' your axes. Homing is the process of moving towards an axis until it hits the switch.

Select the menu option to home each axis in turn: X–, Y–, and Z+. At the end, each axis should be in its appropriate spot, and the machine should be stopped and silent. If it is grinding one axis into a wall, then you need to power off your machine, and start from the beginning of the endstop instructions.

Don't forget to disable your stepper motors: when they get energized they will generate heat and make your motors hot. Turn them off by clicking the 'Disable Steppers' button.

Test Your Extruder

Now your steppers are working, it's time to test your extruder. The extruder testing is pretty simple: you need to make sure everything reports the right temperature, that everything heats up, and that all the motors move in the right direction. Follow the step by step directions and you'll be extruding in no time flat.

All of these instructions will focus on the right half of the control panel which deals with the extruder.

Temperature Measurement Check

First things first: check the temperature readings. These will be in the greyed out 'Current

Temperature' boxes. If you are using the automated build platform, you'll need to check that too. The temperature readings are in Celsius. Since the extruder is off and cold, it should be reading room temperature. Depending on your location and season, that should be something around 21°C.

If your temperature is wildly different from that, then you probably hooked up the wires in the wrong direction. Re-read the Electronics Installation page and double check your wire orientation.

Turn Up The Heat

You should only do this step once your temperature sensors are working! Otherwise BAD THINGS may happen.

To repeat, **NEVER OPERATE THE HEATER** if the reported temperature is anything other than room temperature, usually around 20°C.

First, start off with a low temperature of 50°C. This will allow us to make sure the temperature sensors are truly working. Enter the value into the 'Target Temperature' field. You must tab out or click out of the field for the change to be sent to the machine.

Now, peek into your Thing-O-Matic. There are tiny little LEDs next to each MOSFET. It's a bit tricky to see, but the one near the heater wires should have turned on.

The real test is to wait 10–15 seconds for the temperature to start rising. It should start a consistent up-slope towards 50°C.

Note: If the temperature starts steadily **decreasing**, quickly set the target temperature to 0. It means that you've hooked up your thermocouple backwards! You definitely don't want that. If you see a temperature reversal, you need to shut down the machine, open up the case, and reverse the thermocouple wires.

If you have an automated build platform, test it separately. It should heat up much more quickly as it has a lower thermal mass.

Turn it All the Way Up

Now that your temperature is reporting properly, turn each one up to operating temperature. That is 220°C for the extruder, and 120°C for the heated build surface.

It will take about 5–10 minutes for them to fully heat up and equalize. During that time, the extruder may give off a wee bit of smoke. That is completely normal and is the result of the oils from the anti-sieze burning off. It's a good idea to open a window to let it dissipate.

Control Your Motor Speed

In the meantime, it's time to test your motor! The speed should be set to the default, 1.98. Leave the duration at 10 seconds for the tests (unless you're feeling particularly impatient or meticulous.)

If you're running a MK5 extruder, enter a value of 255 (maximum speed) into the Motor Speed (PWM) field. Tab away from it, and ReplicatorG will now tell your motor what speed it should run at when you turn it on.

Move Your Motor Back and Forth

Next, press the 'Forward' button next to 'Motor Control'. You should hear your motor turn on, and if you visually inspect it from the front, you should see the motor shaft rotating clockwise. Click the reverse button and it should move in the other direction.

If your motor is rotating counter-clockwise in Forward, first double check that you have pressed the 'Forward' button. :) If the Forward button is pressed, that means the motor wires are reversed. Now you need to turn off your machine, open up the electronics bay, and switch the motor wires. This fix works for both DC and stepper motors.

Test Your Conveyor Belt Motor

Note: On Mk6/Mk7 extruders, this will turn the extruder fan on and off. The conveyor motor is controlled by the extruder board on newer ABP.

If you are using the Automated Build Platform, you'll want to test the motor that drives the conveyor belt. Click the '`Build platform belt`' checkbox at the very bottom of the Control Panel.

The motor should whir into life, and your conveyor belt should move with it. The direction of the belt should be towards the front (and the ramp at the front).

If it is not moving at all, or moving in the wrong direction, go back to the electronics wiring step and double check everything. You'll have to turn off the machine and open up the electronics bay for this.

First Extrusion

If everything went according to plan, you should now have your speed set to 1.98 rpm (or a PWM setting of 255 for MK5 users), and your extruder temperature hovering around 220°C. If it has not hit 220°C yet, give it some time. If it has been hovering just below 220°C, bump up the temperature to 225°C for now and you'll need to adjust the PID settings to get it just right.

Make sure your plunger is loosened all the way so you can push the filament in yourself. Take some 3mm filament, and insert it into the extruder, and push it all the way down. If you keep pressing it, you should be able to manually extrude some plastic. Awesome!

Now is time to tighten down the filament plunger. Make it finger tight so the filament is pressed up against the drive gear. Make sure you're at 220°C, and turn your motor on Forward. The filament should begin to be pulled into the extruder, and it should also start extruding.

Plastic is cheap, so let it extrude for at least a couple minutes to 'break in'. Don't get too excited and jump the gun just yet. Make sure it all works properly before going to the next step.

Turn off the Heat

The next step requires a bit of calibration and will take at least 5–10 minutes. It's not a good idea to have your extruder on and hot, so turn it off for now. You can do this by setting the temperature to 0, and clicking the stop motor button.

Thingomatic - How To Print

Level Your Build Platform

In order for your prints to be successful, you need a build surface that is level relative to extruder nozzle. The vast majority of the time, this will just happen as a result of building your machine. Occasionally, you will need to adjust it to make it nice. The flatness of your build surface really only needs to be about +/- 0.5mm due to the 'raft' that will be printed as the first layer. This raft will make up for many irregularities in your build surface.

The basic process is pretty simple:

1. Manually lower your extruder so that it is roughly 1mm above the middle of the platform.
2. Move the build platform so that each corner is under the extruder nozzle in sequence. Note the relative gap height between each corner and the middle.
3. Adjust the height of each corner to compensate for the non-levelness.

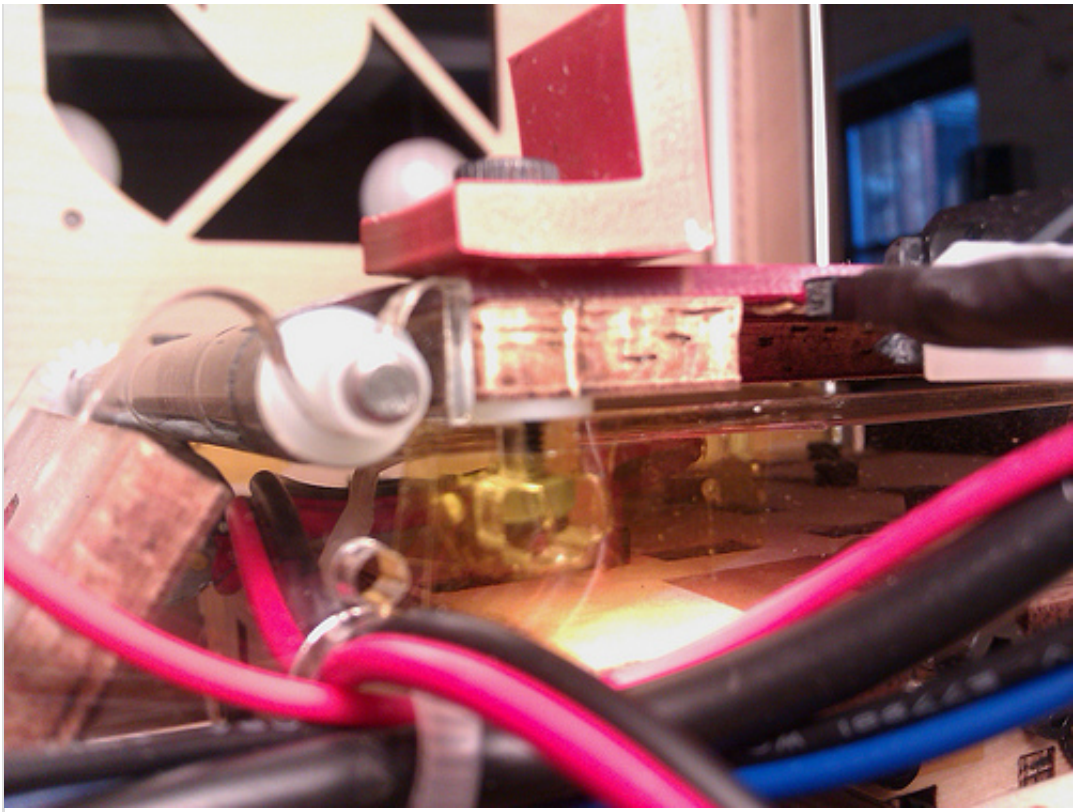
Acrylic Build Platform or Heated Build platform

Leveling the acrylic build platform is very simple. You can loosen the top nuts and adjust the middle nut to move each corner up or down. When you're done, lock the bottom and middle nuts against each other to keep it in place. Do this for all 4 corners and you're done.

Automated Build Platform

The automated build platform does not have a leveling procedure. It your ABP is not quite level, use some 'shims' such as pieces of paper or washers to adjust the height of the ABP platform.

In this picture a plastic washer is used to raise one corner the platform.



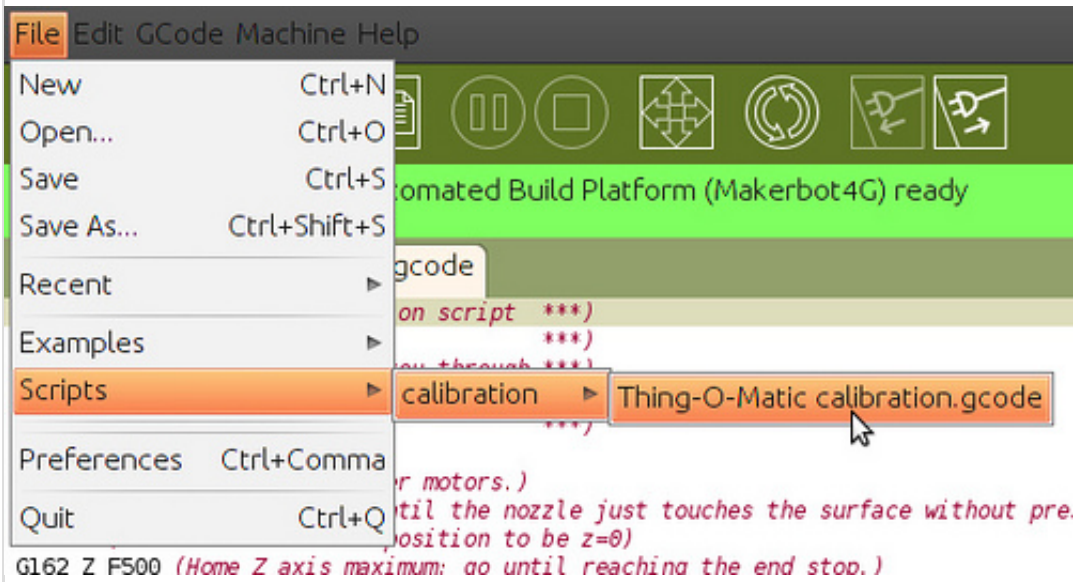
Automatically set your Z-height

Check out this video of the process!

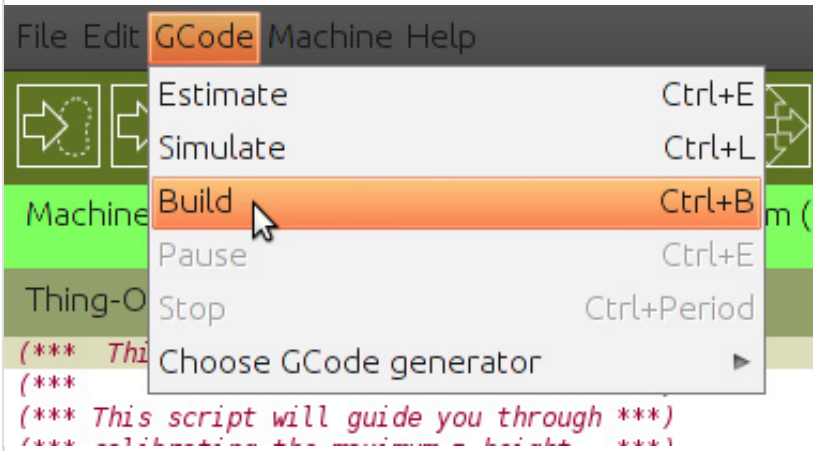
The process for setting your Thing-O-Matic's Z-height has been streamlined with the release of ReplicatorG 25. You no longer need to use a text editor to make changes to the Gcode

profiles.

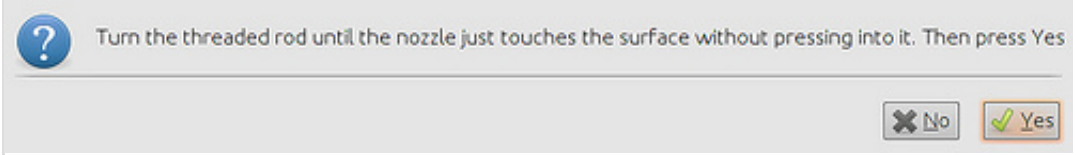
- Open the Thing-O-Matic calibration script: File > Scripts > calibration > Thing-O-Matic calibration.



- Build the file by clicking GCode > Build.



- Turn the threaded rod until the nozzle just touches the surface without pressing into it. Move the build platform around such that the nozzle can reach each of the four corners without catching the belt. If the nozzle catches the belt, relevel your platform or slightly raise your nozzle by turning the threaded rod. Once you are satisfied the nozzle will not catch on the belt, click Okay.



Your machine will now save your Z-height to the motherboard's onboard memory, and you can use the profiles included with ReplicatorG 25 or the new Print-O-Matic method.

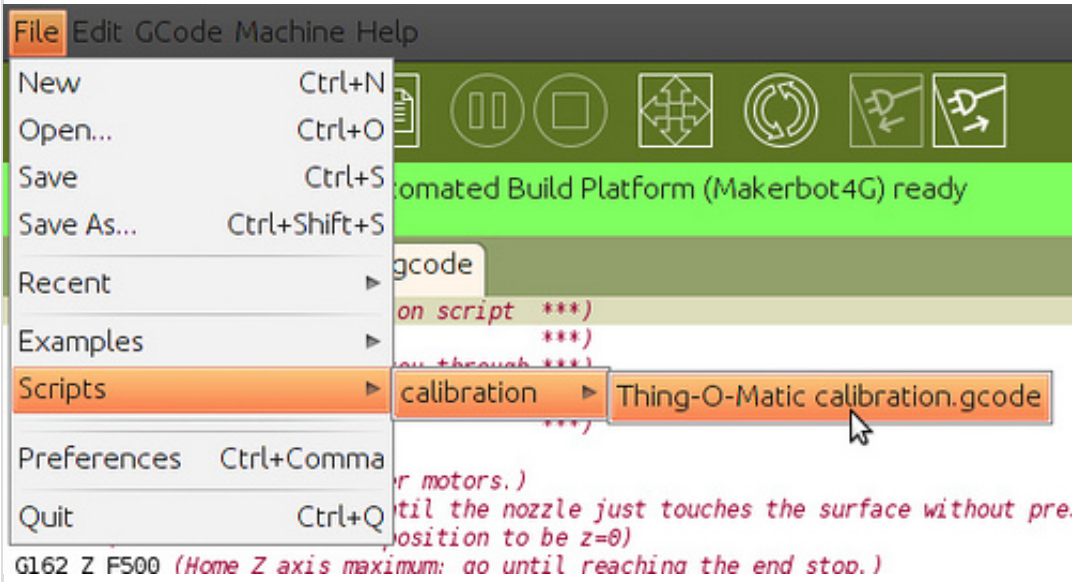
You can now go ahead to the section titled "Get a 3d Model."

Measure Your Z-Height (ReplicatorG 24 and earlier)

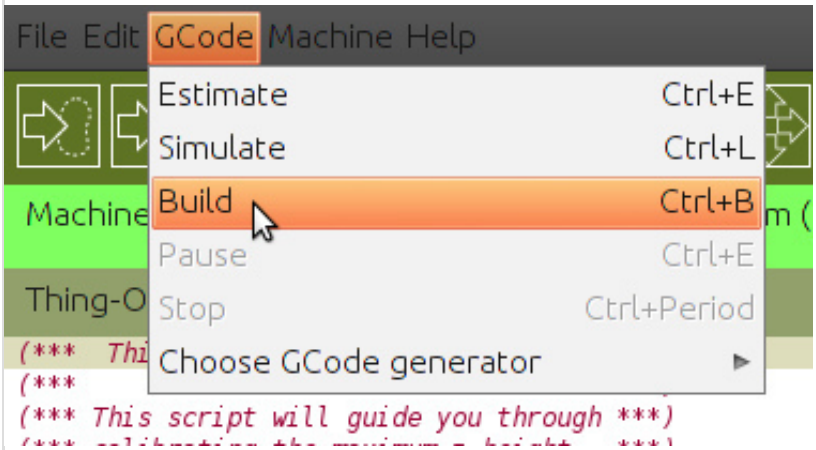
Note: This section is now outdated, use the version above for ReplicatorG 25 and later. This section is preserved for reference only.

The Thing-O-Matic is fully automatic. What that means is that it automatically calibrates its position and start printing the first layer without any sort of intervention by you, the user. Because the exact distance between the Z+ endstop and your build surface can vary from bot to bot, its important to precisely measure it and adjust it if necessary. You encode this information into your build files using the settings you have made for your specific bot.
This process is pretty straightforward:

- Open ReplicatorG 23 or later. Connect to your Thing-O-Matic.
- Open the Thing-O-Matic calibration script: File > Scripts > calibration > Thing-O-Matic calibration.



- Build the file by clicking GCode > Build.

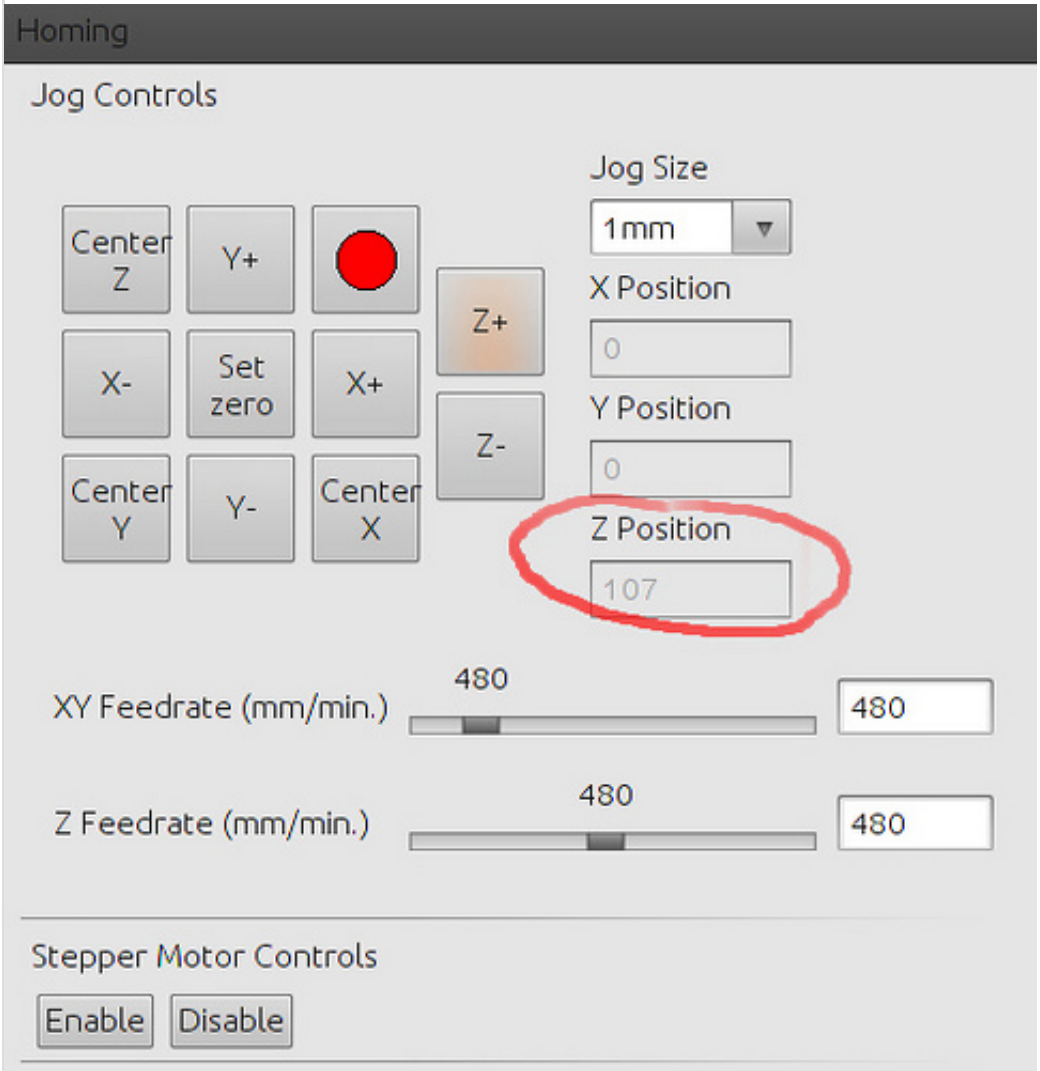


- Turn the threaded rod until the nozzle just touches the surface without pressing into it. Move the build platform around such that the nozzle can reach each of the four corners without catching the belt. If the nozzle catches the belt, relevel your platform or slightly raise your nozzle by turning the threaded rod. Once you are satisfied the nozzle will not

catch on the belt, click Okay.

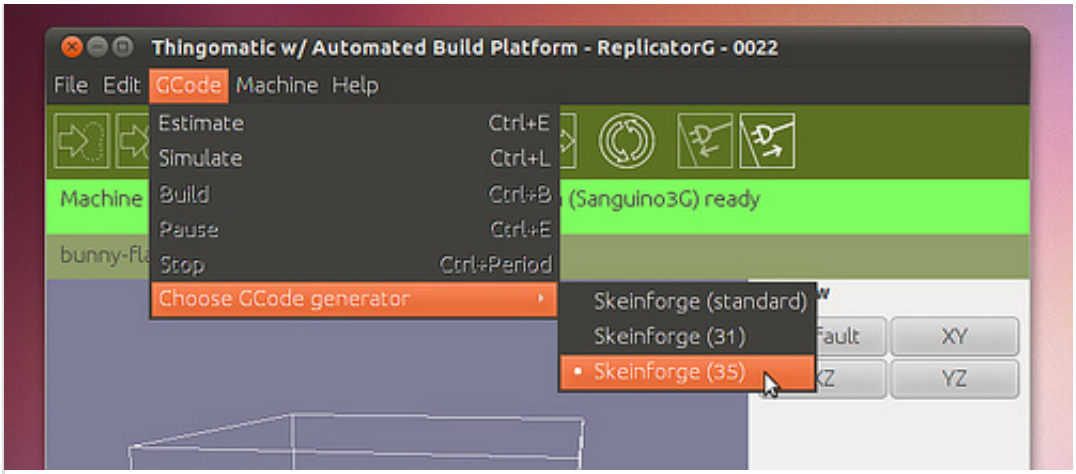
- When the Z platform stops moving click Okay.

- Open the control panel and find "Z Position". This is the "Z axis maximum". This should have a value of approximately 107, each maching being slightly different. Record this value for the next step.



Select GCode Generator

We highly recommend using Skeinforge 35.



Adjust Your GCode Profile

NOTE: With ReplicatorG 0025 it is not necessary to edit the start.gcode file. The axis values will be save to the motherboard and automatically loaded by the existing default start.gcode. If you're using ReplicatorG 25, skip ahead to "Get a 3d model."

With the measured Z distance in hand, you can tell your bot *exactly* how far to go down before starting the build. Warning: if this number is too big, your extruder will RAM itself into the build surface. You do not want that to happen. It is much better to take your number and subtract 1mm from it and do a series of builds where you move closer to the build surface every time than it is to try and get it perfect on the first try and potentially break something.

This is a thing you will only need to do once, so take your time and do it right. Every other build after you get it right will be fully automatic.

This change needs to be made within the directory of files associated with the profile of the Skeinforge edition you will be using to generate your gcode. There is a pair of files in each profile called "start.gcode" and "end.gcode" that establish the behavior during machine warmup (pre-object) and cooldown (post-object) stages respectively. These files will be very powerful for you when you become an advanced MakerBot operator: you can hack your bot to do cool stuff.

For this calibration step, you only need to adjust a number in the "start.gcode" file associated with the profile that tells your machine how to start each build. Check under '`GCode > Choose GCode generator …`' for which edition of Skeinforge you have assigned.

The file you need to edit is located here:

- **Generic file location:** replicatorg-00XX/skein_engines/skeinforge-35/skeinforge_application/prefs/<Profile version>/alterations/start.gcode

You'll need to put in **real values** for replicatorg-00XX and <Profile version>.

So, for example: if you're using ReplicatorG 0024, and the profile for Thingomatic, Stepstruder, Heated Build Platform, and 3mm filament, the file will be at:

- **ReplicatorG 0024 with SF35-Thingomatic-HBP-Stepstruder**: replicatorg-

0024/skein_engines/skeinforge-35/skeinforge_application/prefs/SF35-Thingomatic-HBP-Stepstruder/alterations/start.gcode

If you're using the profile for the Stepstruder with Automated Build Platform and 1.75mm filament, it will be:

- **ReplicatorG 0024 with SF35-Thingomatic-ABP-Stepstruder-1.75mm**: replicatorg-0024/skein_engines/skeinforge-35/skeinforge_application/prefs/SF35-Thingomatic-ABP-Stepstruder-1.75mm/alterations/start.gcode

If you're doing advanced, custom stuff like using skeinforge 39, then you'll need to alter the file in the skeinforge-39 folder, but if that's the case, you're probably on top of this already.

Now, open that file with a plain text editor. On Windows, Notepad++ is a nice one. On Mac or Linux, you can use Bluefish or TextWrangler. In that file, you will see a line that looks like this:

```
G92 Z80 ( ---=== Set Z axis maximum ======= )
```

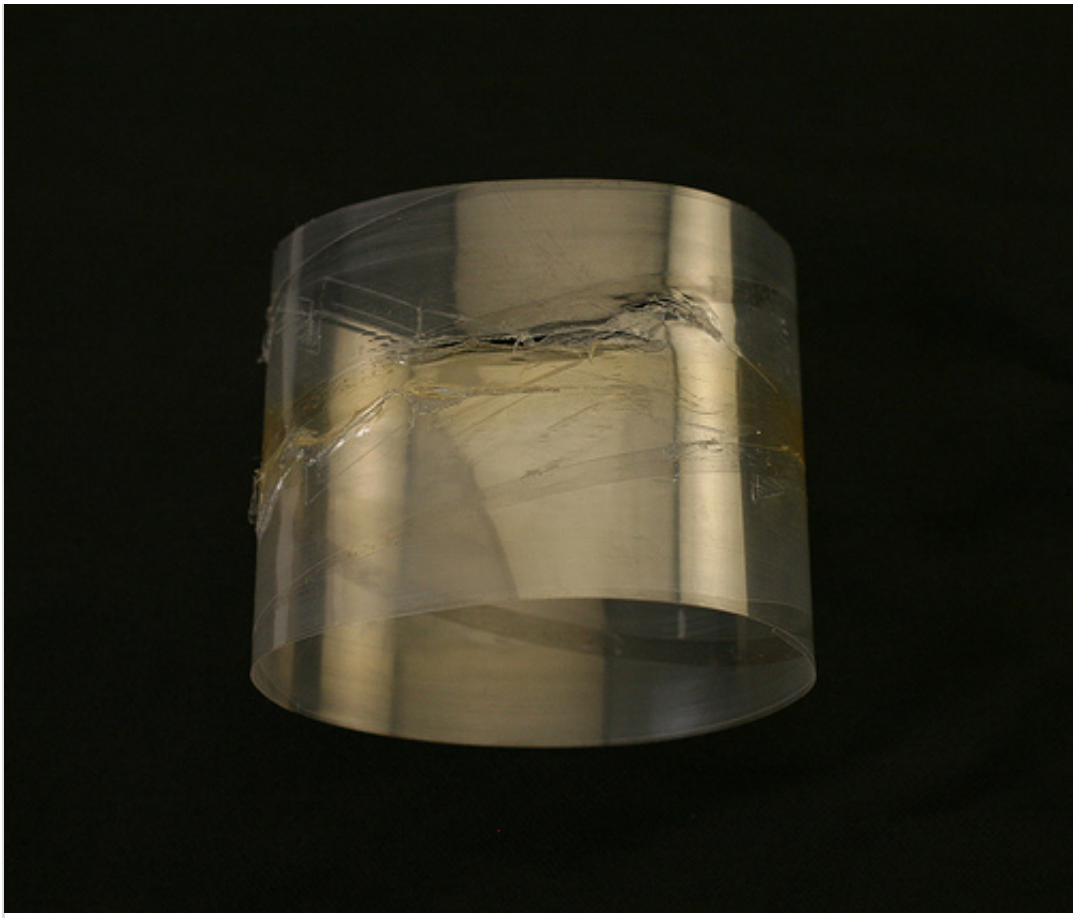
Change this line so that the Z80 number is your measured Z height number. For example, if the number you recorded was 107.2, you will want to modify the line to look like this:

```
G92 Z107.2 ( ------ Set Z axis maximum ===--- )
```

Every time you make a change to this file, you will need to re-slice your model for the changes to take effect.

Be careful, measure it twice and test it carefully.

That way you won't end up with bad prints or a damaged belt like this.



Get a 3D Model

You can either download, design, or 3D scan something to get a 3D model. Downloading is the easiest, and we have created Thingiverse.com as a place for people to share their designs.

We highly recommend starting off with a known good 3D model for your first print. The 20mm calibration cube is a great place to start. Download it to your computer, then open it with ReplicatorG.

Slice Your 3D Model

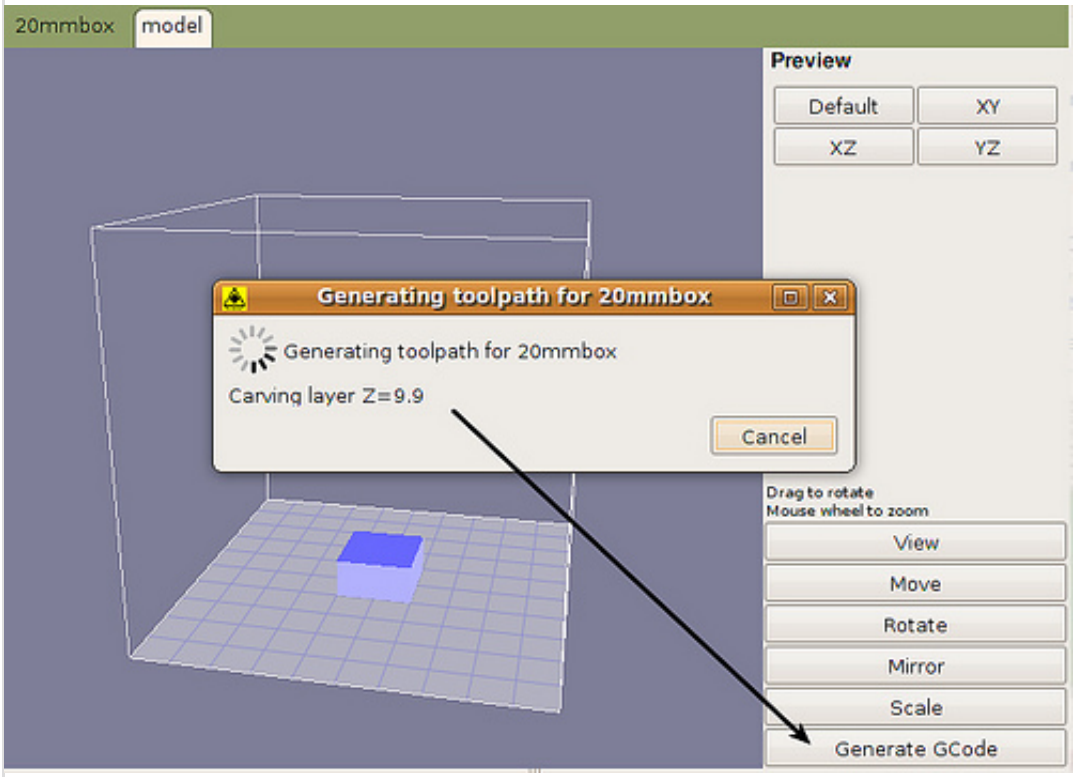
Slicing your model is also known as 'generating the toolpath'. What we are doing here is transforming the 3D model into the exact instructions required to build it. These instructions are in the form of GCODE which is an ancient control language for CNC machines. It is very basic, and tells the robot to go from point to point, when to turn the extruder on or off, and many other things. ReplicatorG then takes these instructions and then tells them to your Thingomatic which follows them and builds you your object.

To slice your object, open it with ReplicatorG, and press the 'Generate GCode' button. Select the 'thingomatic-mk5-automated-abs' profile, click the 'Use Raft' checkbox, and then click 'Ok'.

ReplicatorG will now think for a while. Depending on how complex an object is, and how fast

your computer is, this may take a while. This model is fairly simple, so you can expect it to take anywhere from 1 minute on a fast computer to 5 minutes on a slow computer. Patience is the key here.

Once it completes the toolpath, you're ready to print.



Print Your 3D Model

With the Thing-O-Matic, the printing process is totally automated. Click the Build button, or select 'GCode -> Build' from the main menu. Your MakerBot will go through a warmup routine and then begin building your object. The basic process is this:

- Start warming up the heater
- Home the X, Y, and Z axes
- Move to waiting position to finish warming up (may take up to 10 minutes)
- Extrude for a couple seconds to 'prime the pump'
- Wipe any 'boogers' that will inevitably have formed
- Start printing your object.

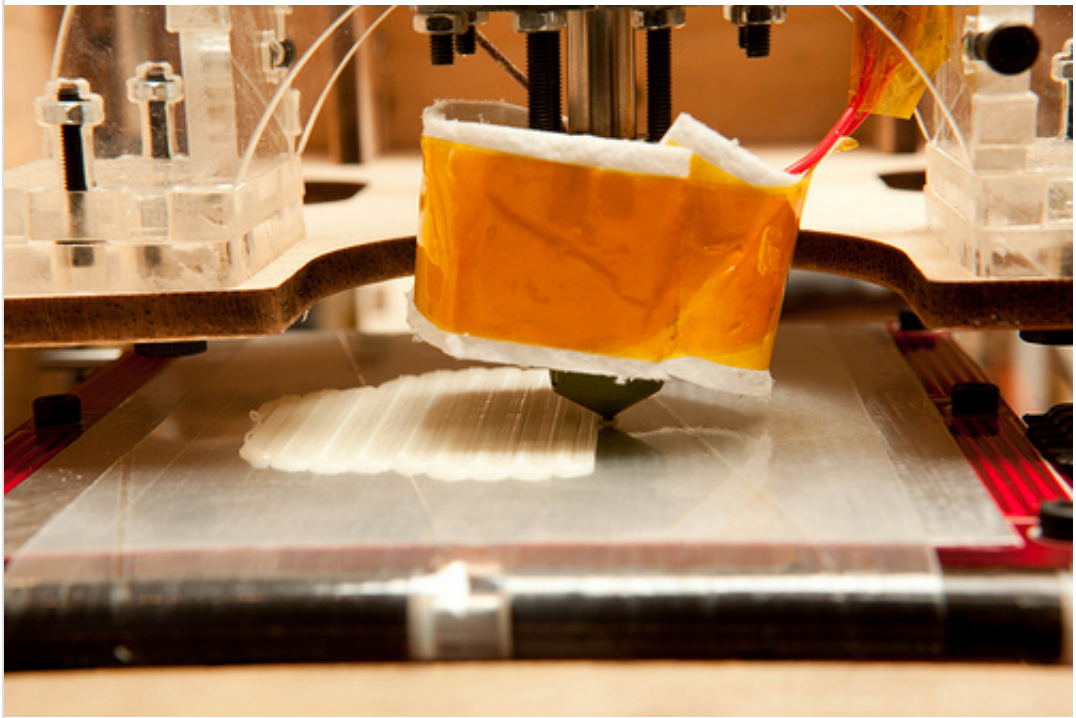
For the first few prints, it's very important to keep an eye on your printer. It may do something weird, or it may have been miscalibrated. Keep one hand on the power switch, and if you sense any sort of danger, flip the switch.

A nice, good looking raft

A raft that is too smooshed

If your raft looks like this, or your nozzle hits the build surface, then your Z height is too large. Decrease the Z-height by an appropriate amount (0.2mm is usually good)

Don't forget to re-slice your object after making this change, otherwise nothing will change. If you do not re-slice, the GCODE that actually makes the printer run will not change, regardless of the start.txt edit.



A raft that is too high

If your raft looks like this or if it is too far off the build surface and is not sticking, your Z height is too small. Increase the Z height by an appropriate amount (0.2mm is good here as well.)

Don't forget to re-slice your object after making this change, otherwise nothing will change. If you do not re-slice, the GCODE that actually makes the printer run will not change, regardless of the start.txt edit.

After the Print

If you are using the Automated Build Platform, it will wait for 90 seconds to let the part cool down, and then it will run the conveyor belt to eject the part. Do not remove the part from the conveyor belt by hand. It will do it for you.

When it is done, it will wipe its nozzle and return to 0,0,0 and be ready for another print.

If you are using the Acrylic Build Platform, you will need some sort of scraper to scrape the object off the surface with. As many CupCake CNC owners can tell you, a sharpened paint scraper works really well! You do not need to wait for the object to cool down, and once its done building your object, your machine will shut down.

If you are using the Heated Build Platform, wait at least 90 seconds for your object to cool down and it should very simple to pull off the build surface.

Maintenance and Upkeep

Keep your Bot Running

The Thing-O-Matic does not require much maintenance. Once you have it built and running, it will provide you with many hours of hands-off printing. However, some of the Thing-O-Matic's mechanical parts, especially the ones in constant movement, are subject to wear and should be tuned from time to time. The following list are things you should be mindful of while using your machine.

Belt Tension

If the belts curve down and the sides flatten, they are likely too loose.

Symptoms of loose belts may include skipped steps, 'backlash' (see below), or infill that doesn't reach a printed object's inner shell surface.

When the motor switches directions, there is a certain amount of movement it must do in order to pick up the slack on the other side before it can actually move the axis. This 'backlash' causes the constructed part to be inaccurate.

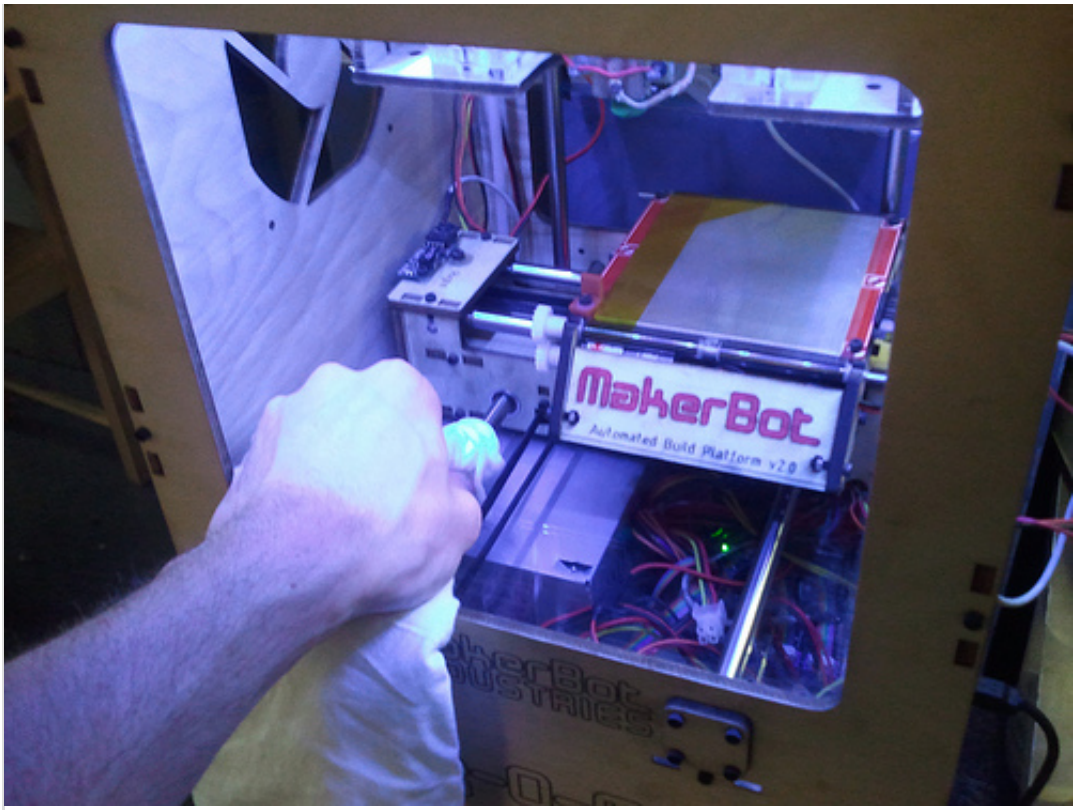
For more information on tightening belts, see the proper belt tensioning page.

Keep X,Y, and Z Rods Lubricated

If your machine is very loud and seems to vibrate a lot when moving, you probably need to lubricate the rods.

All three Thing-O-Matic axes rely on 3/8" diameter precision rods to ensure smooth linear travel without any wobbling. Adding lubrication will clean the rods, reduce friction and minimize long-term wear on the bushings as they slide along these rods.

Grab a clean rag, apply some lubricant to the rag, and run the rag up and down each rod.



Tighten Your Bolts

The bolts on your machine can vibrate loose over time. This is particularly true on the X, Y, and Z axes. Loose bolts can lead to failures or extra noise. If this happens, take your hex key out and tighten any bolts that feel loose.

Upgrade Your Bot

One of the great things about a 3D printer is that it can print its own upgrades, and the Makerbot community is constantly producing designs to improve your bot. You will find a big list of ToM upgrades at http://www.thingiverse.com/thing:6443

Thing-O-Matic Calibration

I need more power, Scotty!

Your Thing-O-Matic is already up and running, but you want to bring the print quality to the next level. Follow these tips, and you will be there in no time.

Strong Stepper Motor Torque

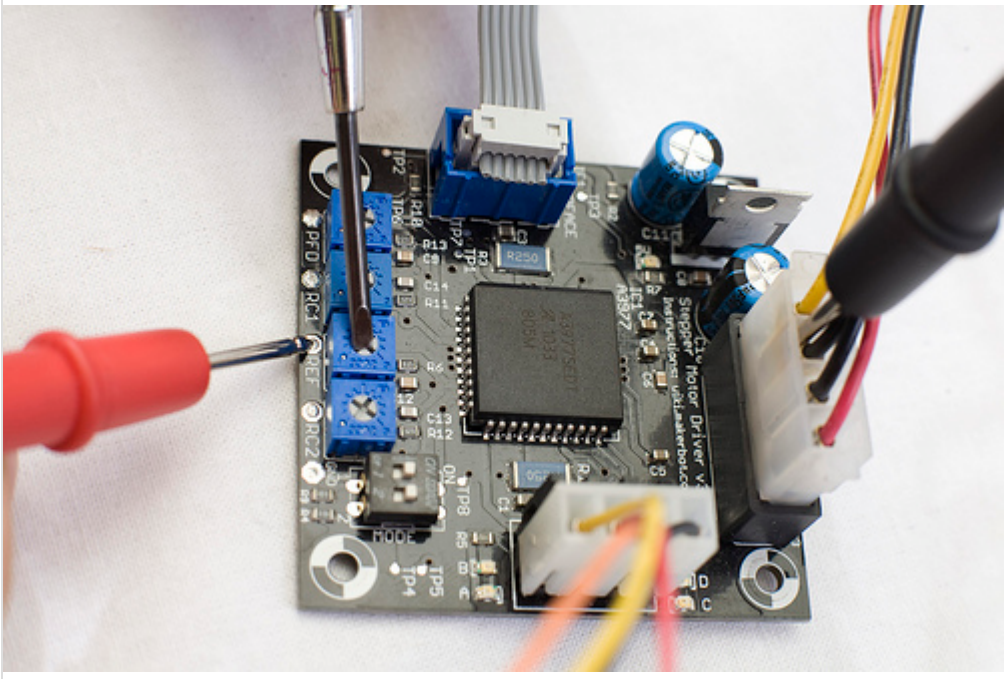
If you've skipped adjusting the pots on your Thing-O-Matic's stepper motor drivers, now is the time to correct that. This can cause a really bad issue: losing steps, even when the bot is otherwise correctly set up.

Losing steps will cause the prints to be oddly shaped and of very poor quality. The obnoxious, grinding noise the stepper motor makes when it can't move is also a telling sign; if this is you, complete the following instructions right now.

Adjusting Stepper Driver Potentiometers

The Generation 4 stepper drivers have four adjustable potentiometers labeled **PFD**, **RC1**, **REF**, and **RC2**. This allows modularity in the electronics so that stepper motors of all different specs and sizes can be controlled with the same driver. A brief description of what each potentiometer controls can be found on the page about the Stepper Driver v3.3

The pot labeled **REF** controls current supplied to the motor. This is the most commonly adjusted pot. Do not adjust the others unless you know what you are doing. Conveniently, each pot has a test point for measuring with a multimeter. Put the positive probe on the test point, and the negative probe to ground.



Note that these values are with the steppers DISABLED, not enabled. 'DISABLED' means that you have not activated your stepper motors in ReplicatorG

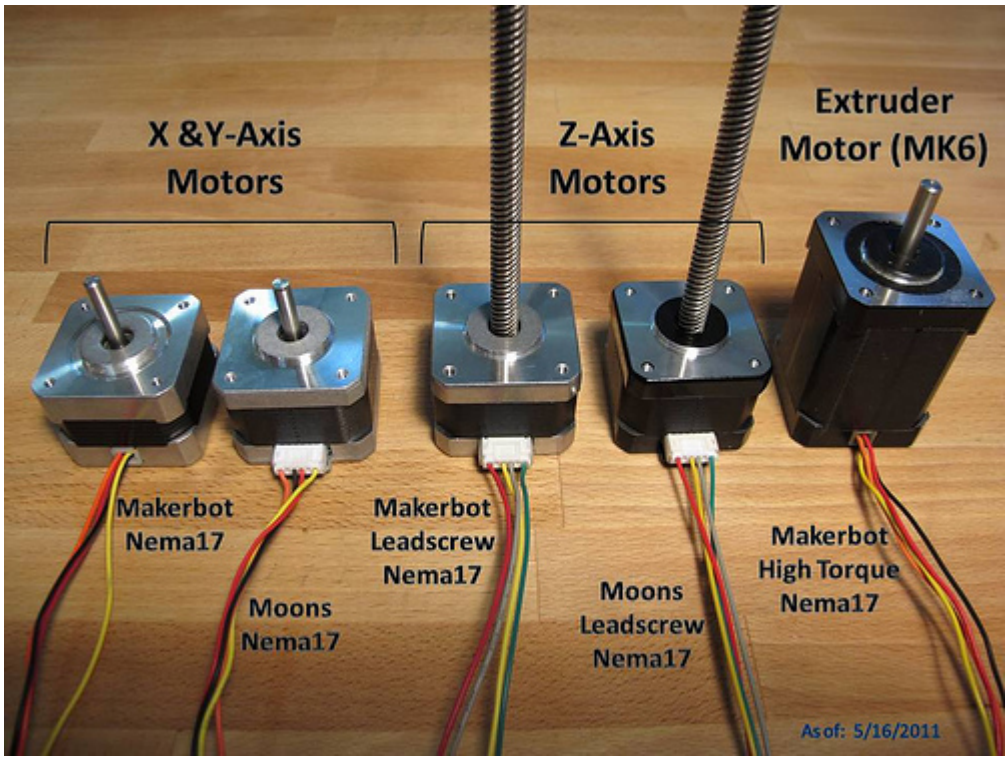
If you do not have a multimeter, counter clockwise rotation will decrease the value, and clockwise will increase for REF and PFD — but it's the **opposite** for RC1 and RC2. There are small indications of the total range for each potentiometer engraved in its case. Suggested values for the stock Thing-O-Matic steppers are:

Stepper Motor	REF Voltage (V)	Rotation between Max and Min
X & Y	0.600	Between 1/8 - 1/4
Z	0.617	Between 1/8 - 1/4

Note: If you set the current too high, the motor will be eventually get **too hot to touch**.

If your Z axis stepper motor is making high pitched noises while set to the above REF value, then you will need to change the other potentiometers as well.

Since the motor suppliers have changed over the course of the Thing-O-Matic's lifetime, you first need to identify which motors you have from the image below, and then adjust their respective potentiometers accordingly.



The suggested pot values for the **Z-axis stepper motors ONLY**:

Stepper Motor	PFD (V)	RC1 (V)	REF (V)	RC2 (V)
Makerbot Leadscrew Nema17	2.311	0.952	0.617	0.963
Moons Leadscrew Nema17	2.311	0.952	0.617	0.963

The suggested pot values for the **X&Y-axis stepper motors ONLY**:

Stepper Motor	PFD (V)	RC1 (V)	REF (V)	RC2 (V)

Makerbot Nema17	1.952	0.953	0.600	0.955
Moons Nema17	1.952	0.953	1.68	0.955

The suggested pot values for the **Stepstruder MK7 stepper motor ONLY**:

Stepper Motor	PFD (V)	RC1 (V)	REF (V)	RC2 (V)
Stepstruder MK7	1.952	0.953	1.68	0.955

The suggested pot values for the **Stepstruder MK6 stepper motor ONLY**:

Stepper Motor	PFD (V)	RC1 (V)	REF (V)	RC2 (V)
Stepstruder MK6	2.31	0.94	1.56	0.94

For those of you who want to go a little deeper, here's a table that explains a little bit more about the properties of these motors.

	Motor Type	REF Voltage (V)	Current Through Motor (A)	Theoretical Current Limit (A)	Winding Resistance (Ohms)
X&Y Axis	Makerbot Nema17	0.6	0.3	0.34	35
	Moons Nema17	1.68	0.84	2.09	5.75
Z Axis	Makerbot Leadscrew Nema17	0.617	0.3085	0.39	31
	Moons Leadscrew Nema17	0.617	0.3085	0.40	30
Extruder	Makerbot High Torque Stepper	1.56	0.78	0.92	13

Extruder Temperature

Ideally, the goal is to use the lowest temperature which produces the desired results.

If the temperature is too high, this can lead to drips and sagging lines. These are especially noticeable on overhangs. If the temperature is too low, it can lead to poor adhesion, delamination, and edge curling.

By default ReplicatorG 0025 sets the MK6+ Extruder temperature to 225C. However, this will occasionally need to be adjusted for optimum results. Often, when using a new batch of filament, the temperature will need to be adjusted. This can be due to variations in the filament diameter, absorbed moisture levels, and even color used.

3mm ABS Filament by Color	Extruder Temp
Natural	225C
Black	230C
Nuclear Green	235C

Note: These are starting points for reference only. Each Thing-O-Matic build and each batch of filament may have individual variations.

Establish Ideal PID Settings

If the extruder seems to have trouble reaching or maintaining its target extrusion temperature, you may need to adjust the heater's control system parameters.

The Thing-O-Matic maintains its operating temperature with a feedback loop, and the PID settings will mathematically control how the system reacts to that feedback. A very detailed explanation of the theory behind how this works can be found on Wikipedia's article about PID Controllers

Default PID Settings For Plastruder MK5

To edit the PID settings, you need to have your machine connected to a computer with ReplicatorG running. Under the **Machine** file menu, select **Toolhead Onboard Preferences**.

The recommended defaults are listed below:

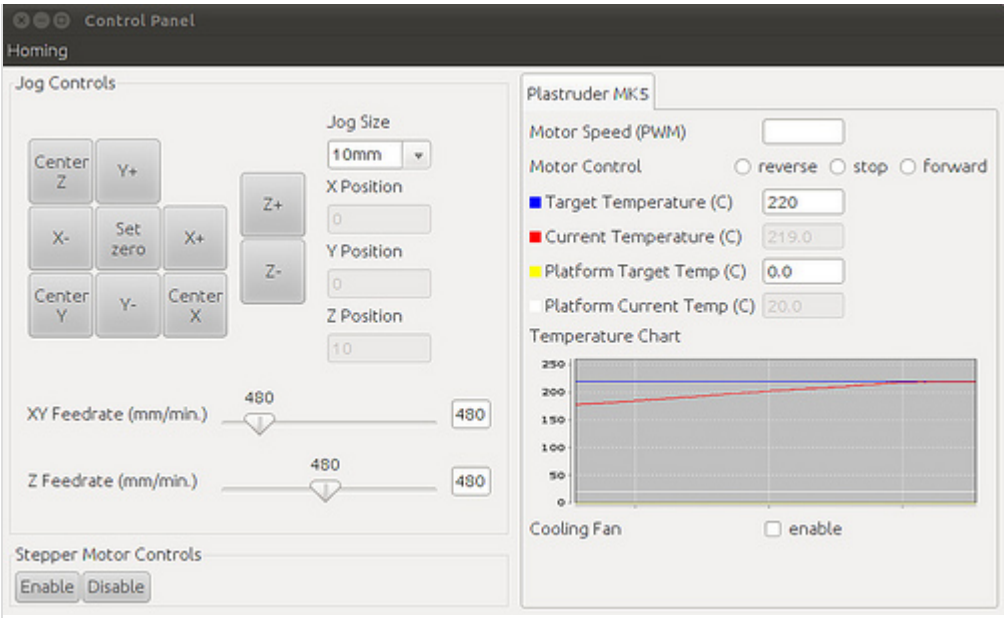
P : 6.25
I : 0.4140625
D : 100.0

While the default settings will work well for the vast majority of users, it is still recommended that you verify they are working well. ReplicatorG has made this very easy to do with its built-in **Temperature Chart** in the control panel.

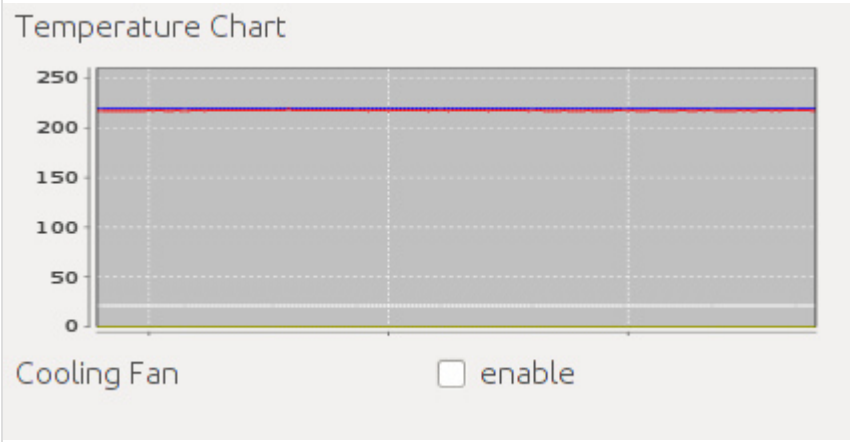
While in the control panel:

1. Set the **Target Temperature (C)** field value to **220**
2. Pressing Enter on the keyboard.

The chart will then display a **blue** line indicating the target temperature, and for the next couple of minutes you can watch the **red** line to see how the extruder heats up. The Temperature chart below demonstrates what you want to see:



Over long periods of time the **red** line on the chart should settle directly on or within a degree of your set **blue** line target temperature:



You will know when the PID settings are correct because it will allow the system to rise to the set target temperature at a relatively constant rate, have very little to no overshoot, and will settle over time on the target temperature without much fluctuation.

Manually Adjusting the PID Settings

If your Temperature Chart does not look like the one above, it is likely you need to tweak one or more of the PID parameters. Depending on how your system is acting up, you can choose which parameter is best to change by using both the parameter chart and labeled image found below.

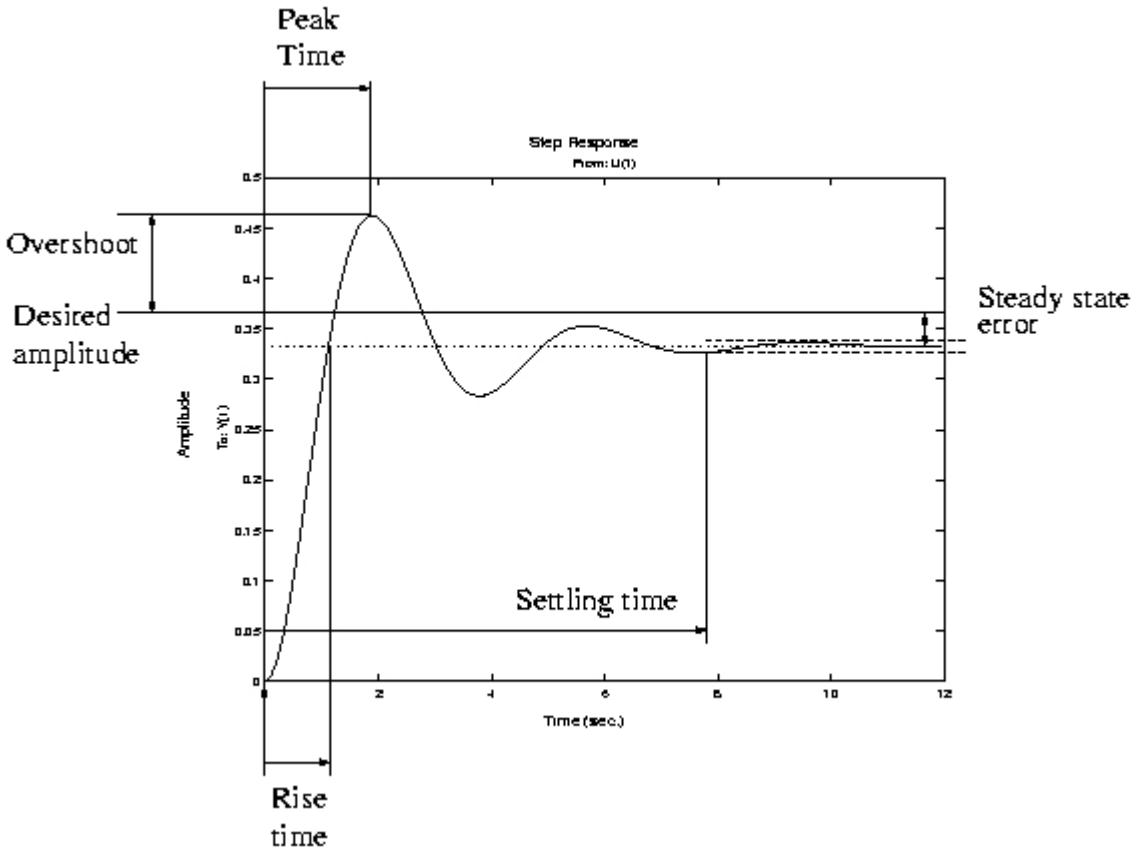


Image via University of Notre Dame Aerospace and Mechanical Engineering

Effects of *increasing* a parameter independently

Parameter	Rise time	Overshoot	Settling time	Steady-state error	Stability[3]
K_p	Decrease	Increase	Small change	Decrease	Degrade
K_i	Decrease[4]	Increase	Increase	Decrease significantly	Degrade
K_d	Minor decrease	Minor decrease	Minor decrease	No effect in theory	Improve if K_d small

Chart via Wikipedia

The parameter chart describes howeach visual characteristic on your Temperature Chart will be effected by an INCREASE in the P, I, or D parameter.(Labeled Kp,Ki,Kd)

The image labels howthese characteristics would be apparent in your Temperature Chart.

For example, if you are not able to reach your set target temperature, the image shows you'll want to decrease the steady-state error, and the parameter chart shows this can be done by either increasing P, or increasing I by a very small amount.

As another example, if you are seeing temperature oscillations by more than a couple degrees above and below your target temperature, you'll want to improve stability, and the parameter chart shows us that this can be resolved in most cases by increasing D.

It is important to understand that **this is very much a guess-and-check process**, yet you shouldn't need to tweak your settings very far away from the defaults. After making a parameter change, it is always required that the Extruder Board is **reset** for the changes to take effect.

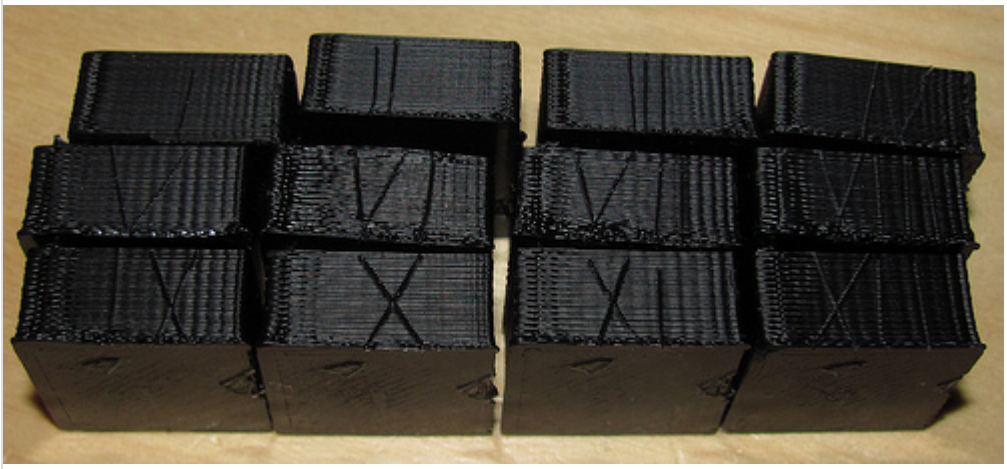
Improving the Default Skeinforge Model Slicing Profile

The default Thing-O-Matic Skeinforge profile that is packaged with ReplicatorG should produce some nice looking parts out of the box, but you want to be certain that your settings are optimized for your Thing-O-Matic.

Basic Tuning Process

Please note: this refers to an outdated script. We're working on documenting our new Print-O-Matic feature in ReplicatorG 25 and beyond! This is here for reference purposes only.

The basic procedure for tuning your skeinforge profile is to change one setting at a time, do a test print, examine the output, and repeat. There are some quirks to the process, but that is about it. Before you get started, we highly recommend you read this awesome guide by MakerBot Operator Dave Durant. It will help you tremendously.



Paraminator.py - Automated Testing of Parameters

The process above is a fairly intensive process. For every single change, you have to re-slice your object, print it, mark it, and then do it all over again. This can be pretty laborious, not to mention the process can take many hours.

With the advent of the automated build platform, we can now automate a portion of that process. We've written a script called **paraminator.py** which allows you to pick one parameter in skeinforge, choose a minimum and maximum value, and it will generate a single gcode file that contains a series of prints that each have been sliced with a different value for the chosen parameter.

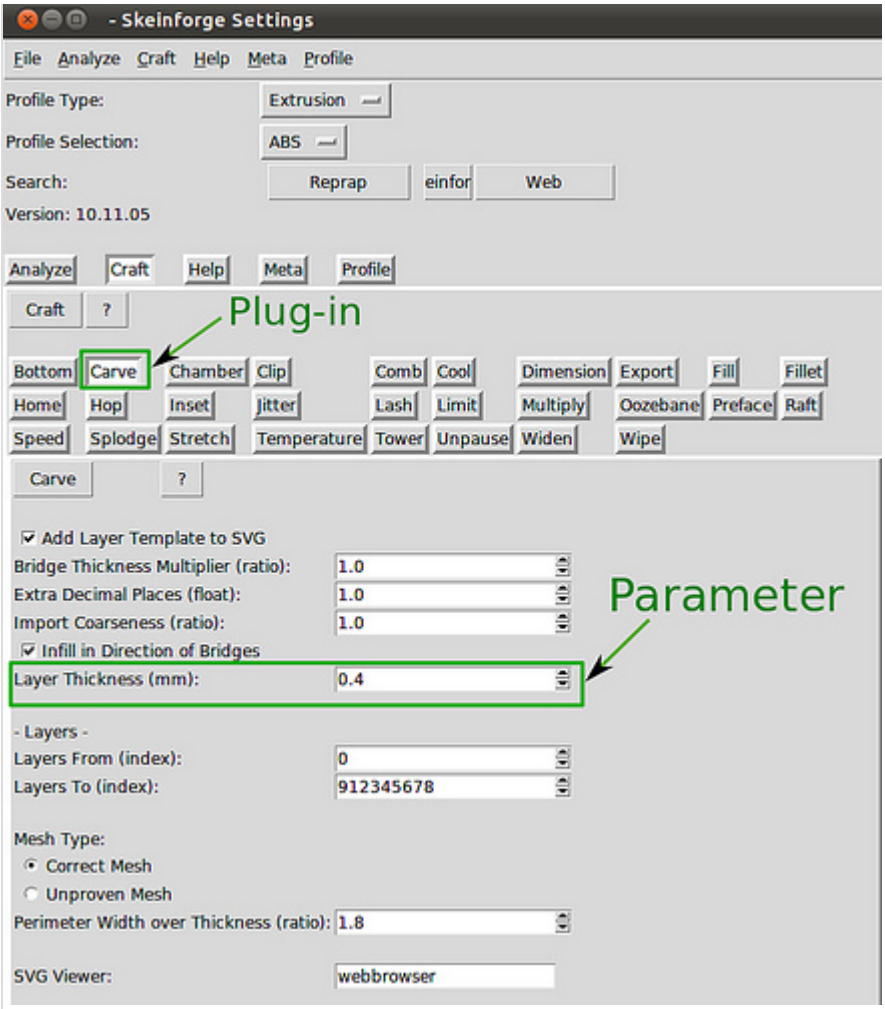
In order to help you identify the objects after they have printed, the objects to be printed are sequential block letters with the embossed letter on top. The idea is that the user looks at the printed letters, and decides which iteration has the best visual results. Looking back at the gcode, it is specified which letter block corresponds to each value, and the best performing value can be input into skeinforge. Using this process with any skeinforge parameters in question will allow you to tune you machine to the max!

How to Use Paraminator.py

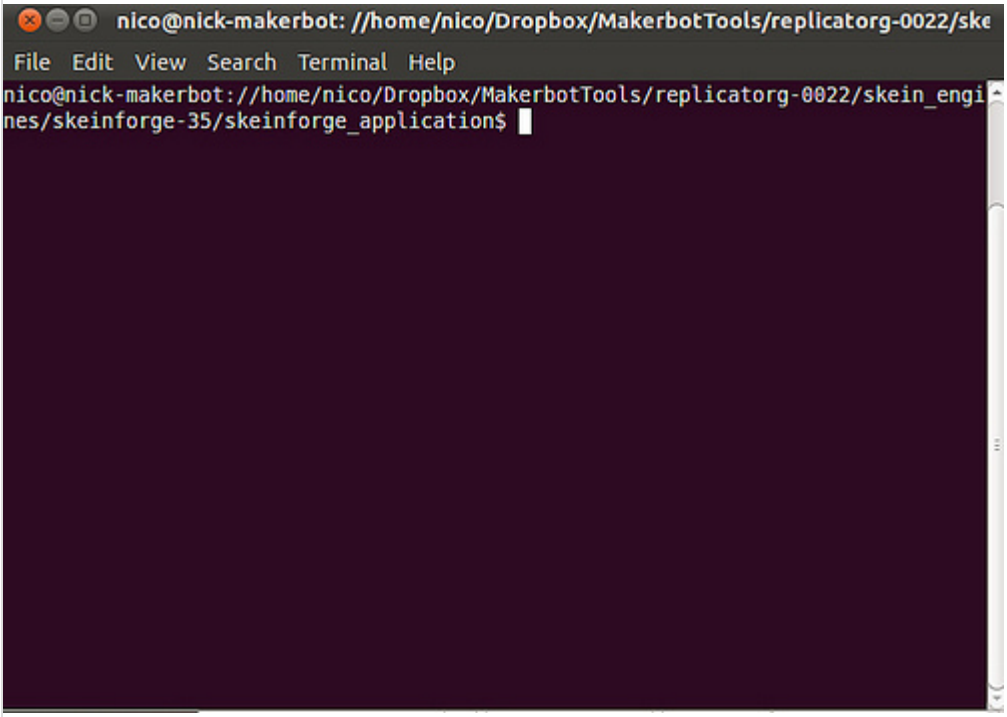
Currently, the paraminator script is only functional via the command line. First verify that "*paraminator.py*" and the "*test-objects*" folder are both living in the **ReplicatorG/skein_engines/skeinforge-35/skeinforge_application** directory. The test-objects folder should contain 26 letter block stls.

You will also need python to be in your path: this means that if typing "python" doesn't start a python prompt (which, btw, can be exited by typing in the command "quit()" if yours works) it will have to be added. Linux and OS X are probably OK, but windows users will likely need to edit their "Environment Variables" in the "Advanced" section of "System Properties," which can usually be accessed by right-clicking "Computer" or "My Computer" and choosing "Properties." Add the name of the folder where python is installed, often C:\Python27 or something similar.

Now decide which setting in skeinforge you would like to explore, and take note of exactly how it appears in skeinforge, and which plug-in it belongs to. For instance, if you want to change the layer thickness, it appears in skeinforge as "Layer Thickness (mm)" (capitalization matters) and it is part of the Carve plug-in.


- Skeinforge Settings
File Analyze Craft Help Meta Profile
Profile Type: Extrusion
Profile Selection: ABS
Search: Reprap einfor Web
Version: 10.11.05
Analyze Craft Help Meta Profile
Craft ?
Plug-in
Bottom Carve Chamber Clip Comb Cool Dimension Export Fill Fillet
Home Hop Inset Jitter Lash Limit Multiply Oozebane Preface Raft
Speed Splodge Stretch Temperature Tower Unpause Widen Wipe
Carve ?
Add Layer Template to SVG
Bridge Thickness Multiplier (ratio): 1.0
Extra Decimal Places (float): 1.0
Import Coarseness (ratio): 1.0
Infill in Direction of Bridges
Layer Thickness (mm): 0.4
Parameter
- Layers -
Layers From (index): 0
Layers To (index): 912345678
Mesh Type:
Correct Mesh
Unproven Mesh
Perimeter Width over Thickness (ratio): 1.8
SVG Viewer: webbrowser

Open up the Terminal, and navigate to the **ReplicatorG/skein_engines/skeinforge-35/skeinforge_application** directory.



The layer thickness test example will be carried out through the entire paraminatin' process. Before the script is run, there are 6 things that need to be specified:

- **file** - this is where you locate the plug-in file. For the example above, the carve.csv file would need to be specified. In skeinforge 35, the csv files are located deeper within the skeinforge_application directory at: **prefs/SF35-Thingomatic-ABP/profiles/extrusion/ABS/carve.csv**

- **parameter** - this is the parameter that is going to be changed, and it needs to be written with quotations exactly how it appears in skeinforge: **"Layer Thickness (mm)"**

- **start** - this is the starting value for the parameter you want to explore. It's a good idea to have a starting value below your expected ideal value, so that it will be clear when you reach the ideal value. For the layer height test, we will choose: **0.32**

- **end** - this is the ending value for your parameter. It should be higher than expected. For the layer height test, we will choose: **0.40**

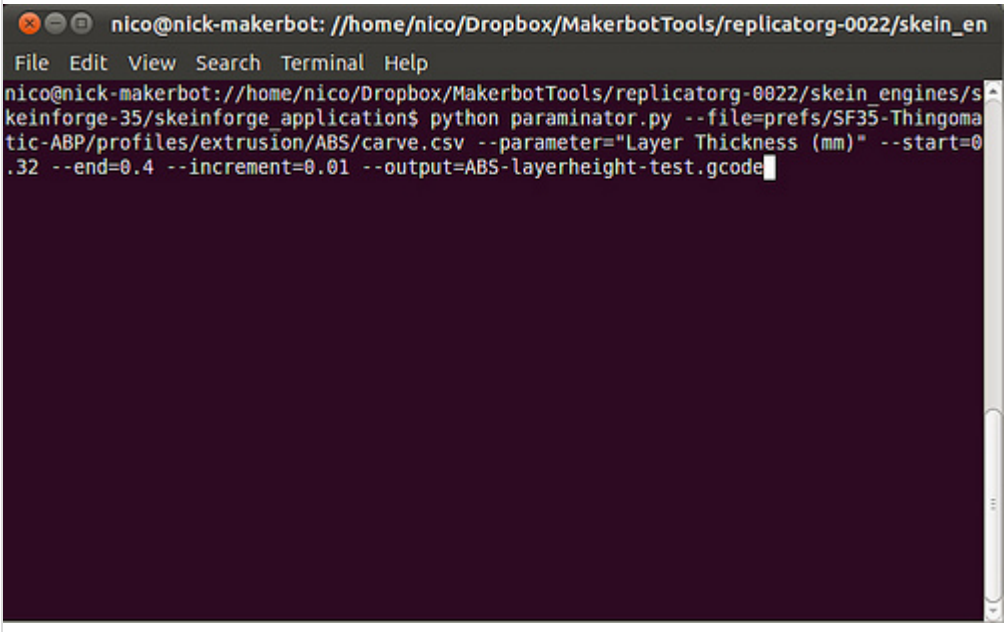
- **increment** - this dictates how many objects will be printed in your test. For the layer height test, if an increment of 0.04 is chosen, it would create 3 objects: 0.32, 0.36, and 0.40. If we were to choose an increment of 0.02, it would create 5 objects: 0.32, 0.34, 0.36, 0.38, and 0.40. To be thorough, lets choose an increment of **0.01**.

- **output** - this is where you specify the file name you want for the test. The output will be a single gcode file that queues up all the incremental parts into a single print session on the ABP. The output gcode file can also be found in the skeinforge_application directory. For the layer height test, we will choose our output name as: **ABS-layerheight-test.gcode**

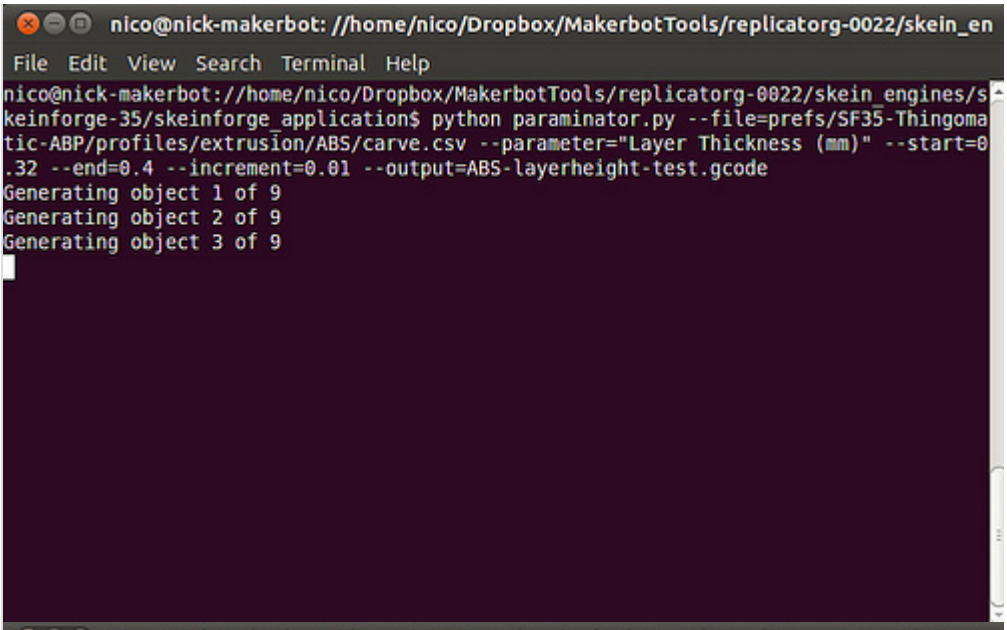
Now go back to the terminal, and get ready to execute the paraminator script. The six specifications above will all be typed into the command line to execute the script:

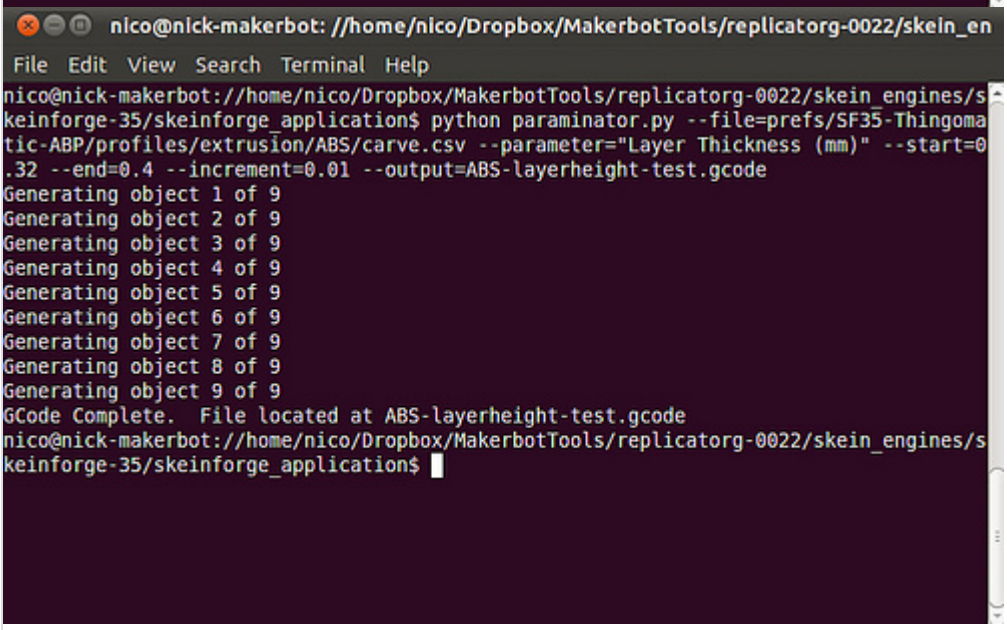
python paraminator.py --file=prefs/SF35-Thingomatic-ABP/profiles/extrusion/ABS/carve.csv --parameter="Layer Thickness (mm)" --start=0.32 --end=0.4 --increment=0.01 --output=ABS-layerheight-test.gcode

Note: the "--" above is actually two hyphens

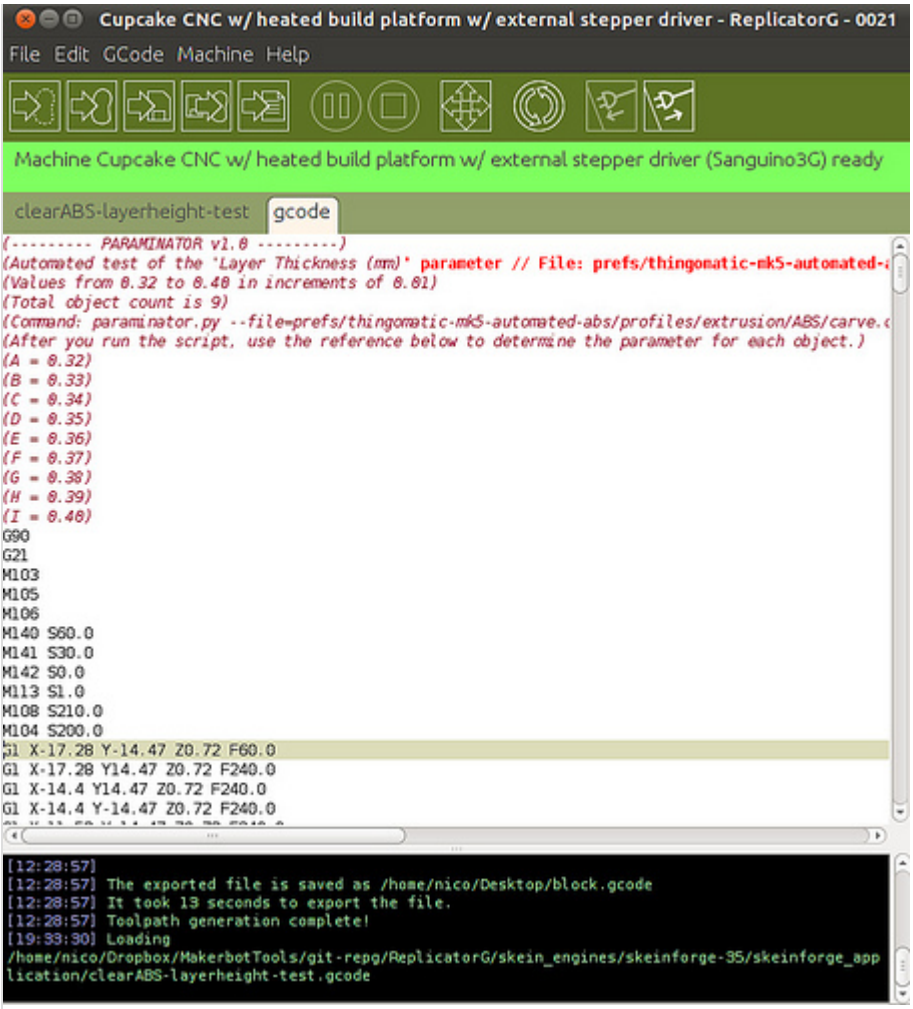

nico@nick-makerbot: //home/nico/Dropbox/MakerbotTools/replicatorg-0022/skein_en
File Edit View Search Terminal Help
nico@nick-makerbot://home/nico/Dropbox/MakerbotTools/replicatorg-0022/skein_engines/skeinforge-35/skeinforge_application$ python paraminator.py --file=prefs/SF35-Thingomatic-ABP/profiles/extrusion/ABS/carve.csv --parameter="Layer Thickness (mm)" --start=0.32 --end=0.4 --increment=0.01 --output=ABS-layerheight-test.gcode

The script will start generating your objects.


nico@nick-makerbot: //home/nico/Dropbox/MakerbotTools/replicatorg-0022/skein_en
File Edit View Search Terminal Help
nico@nick-makerbot://home/nico/Dropbox/MakerbotTools/replicatorg-0022/skein_engines/s
keinforge-35/skeinforge_application$ python paraminator.py --file=prefs/SF35-Thingoma
tic-ABP/profiles/extrusion/ABS/carve.csv --parameter="Layer Thickness (mm)" --start=0
.32 --end=0.4 --increment=0.01 --output=ABS-layerheight-test.gcode
Generating object 1 of 9
Generating object 2 of 9
Generating object 3 of 9


nico@nick-makerbot: //home/nico/Dropbox/MakerbotTools/replicatorg-0022/skein_en
File Edit View Search Terminal Help
nico@nick-makerbot://home/nico/Dropbox/MakerbotTools/replicatorg-0022/skein_engines/s
keinforge-35/skeinforge_application$ python paraminator.py --file=prefs/SF35-Thingoma
tic-ABP/profiles/extrusion/ABS/carve.csv --parameter="Layer Thickness (mm)" --start=0
.32 --end=0.4 --increment=0.01 --output=ABS-layerheight-test.gcode
Generating object 1 of 9
Generating object 2 of 9
Generating object 3 of 9
Generating object 4 of 9
Generating object 5 of 9
Generating object 6 of 9
Generating object 7 of 9
Generating object 8 of 9
Generating object 9 of 9
GCode Complete. File located at ABS-layerheight-test.gcode
nico@nick-makerbot://home/nico/Dropbox/MakerbotTools/replicatorg-0022/skein_engines/s
keinforge-35/skeinforge_application$

The script is finished and you can now open the gcode in ReplicatorG, and start testing! In the beginning of the gcode, there is a commented out reference table for determining which value corresponds with each letter block.


Cupcake CNC w/ heated build platform w/ external stepper driver - ReplicatorG - 0021
File Edit GCode Machine Help
Machine Cupcake CNC w/ heated build platform w/ external stepper driver (Sanguino3G) ready
clearABS-layerheight-test
gcode
(--------- PARAMINATOR v1.0 ---------)
(Automated test of the 'Layer Thickness (mm)' parameter // File: prefs/thingomatic-mk5-automated-
(Values from 0.32 to 0.40 in increments of 0.01)
(Total object count is 9)
(Command: paraminator.py --file=prefs/thingomatic-mk5-automated-abs/profiles/extrusion/ABS/carve.
(After you run the script, use the reference below to determine the parameter for each object.)
(A = 0.32)
(B = 0.33)
(C = 0.34)
(D = 0.35)
(E = 0.36)
(F = 0.37)
(G = 0.38)
(H = 0.39)
(I = 0.40)
G90
G21
M103
M105
M106
M140 S60.0
M141 S30.0
M142 S0.0
M113 S1.0
M108 S210.0
M104 S200.0
G1 X-17.28 Y-14.47 Z0.72 F60.0
G1 X-17.28 Y14.47 Z0.72 F240.0
G1 X-14.4 Y14.47 Z0.72 F240.0
G1 X-14.4 Y-14.47 Z0.72 F240.0
[12:28:57]
[12:28:57] The exported file is saved as /home/nico/Desktop/block.gcode
[12:28:57] It took 13 seconds to export the file.
[12:28:57] Toolpath generation complete!
[19:33:30] Loading
/home/nico/Dropbox/MakerbotTools/git-repg/ReplicatorG/skein_engines/skeinforge-35/skeinforge_application/clearABS-layerheight-test.gcode

Good luck, and happy paraminating!

How to Print Raftless

Overview

Raftless printing is an advanced technique you can use to reduce waste, improve the quality of your parts, and generally improve your MakerBot experience. It can be a bit tricky to setup, but once you have it working it is simple to use.

The basic strategy of raftless printing is to print the first outline of the first layer at a much slower speed than normal. This allows even the most difficult outlines to stick to the build surface. Once the first outline is drawn, the rest of the object is built. Getting the first outline to stick 100% is the most critical part of the build and if that fails, then the entire print will fail. It's literally all or nothing.

Step 1: Create a new Profile

You will want to be using the latest version of ReplicatorG, and the latest Skeinforge included in ReplicatorG. As of this writing, that is ReplicatorG 0023 and Skeinforge 35.

This guide assumes you have a properly calibrated and functioning Skeinforge 35 profile. If you do not have your printer working nicely with a raft, then you need to do that first. Only attempt this once you are printing nice models with a raft.

Create a new profile by opening ReplicatorG, opening a 3D model if you need to, and then clicking the 'Generate GCode' button.

This pulls up the 'Choose a skeinforge profile' window. Select the profile you currently use for slicing, and then click 'Create…'. Give your profile a descriptive name and click 'OK'. The Skeinforge configuration screen will then pop up to allow you to modify your profile.

Step 2: Enable Outline

Since raftless printing requires a perfect first outline, we've included the Outline plugin in Skeinforge which draws a box around your object for the first layer. This box 'primes the pump' and gets your extruder in the perfect state for printing the first layer.

Click on the Outline button and make sure the 'Activate Outline' box is checked.

You may experience that you obtain best results by disabling the 'Wipe' checkbox in the 'Wipe' panel.

Step 3: Update your Raft settings

Raftless printing is made possible by adjusting the settings in the Raft plugin. We will still be using the plugin itself, just not to generate a Raft.

1. Set the "Base Layers (integer)" option to 0 (zero).

2. Set the "Interface Layers (integer)" option to 0 (zero).
3. Set the "Object First Layer Feed Rate Infill Multiplier (ratio)" to 1.25.
4. Set the "Object First Layer Feed Rate Perimeter Multiplier (ratio)" to 0.5.

Step 4: Test Print!

Printing with a Raft is simple, and the raft itself will self correct for Z height irregularities. This is why it is the option we recommend to new operators. Typically, the starting Z height for a raft build is higher than the starting Z height for a raftless build.

Now that you have enabled raftless, its time for a test print. Raftless printing requires you to have your Z height calibrated exactly. The easiest way to do this is to simply print test objects until you have it working.

Coming from a raft based print, your initial Z height will be too high. This is okay, because you need a baseline to work from. Your first test object is pretty much guaranteed to fail, so don't worry. This first print is to gather data so you know how to tweak your build.

In order to fully test your raftless setup, I highly recommend printing an object that has a complex outline geometry. I highly recommend that you use a gear as a test print object. The teeth means your extruder will constantly be changing direction and if the first layer is going to rip up, it will rip up with the gear.

Once you can print a gear flawlessly with Raftless, you'll be able to print anything with raftless.

Step 5: Adjust your Z height

After you have printed the first test object, you may need to adjust your Z height. If you have your Z height set too high on a raftless build, the print will simply not stick and you'll need to try again.

If your Z height is too low, it is very dangerous. You can easily run the nozzle into the build surface and damage it. Be very careful and slowly work your way towards the ideal Z height. It is very important that you be careful.

For testing purposes, it's very easy to adjust your Z height: find the line in your GCode file that looks something like this:

G92 X-57.5 Y-60 Z108.2 (set zero)

This line is what tells ReplicatorG what the current position of the extruder nozzle is after it has moved to each endstop. Changing the Z number will make ReplicatorG start the print at a different height. This parameter is very dangerous. A typo can easily slam the extruder into the build surface.

Also, this parameter is not quite intuitive. Because the numbers tell ReplicatorG what its current position is, changing the numbers has a non-intuitive effect. To be more specific, here is the effect of changing the numbers:

INCREASING the Z, will set the starting layer height LOWER because ReplicatorG thinks its starting from a higher distance.

DECREASING the Z will set the starting layer height HIGHER because ReplicatorG thinks its starting from a lower distance.

Therefore, when you adjust your starting distance DOWN to get your layers to stick, you will need to slightly INCREASE the Z number. I prefer to adjust that number in increments of 0.2mm.

Step 6: Test and Repeat

You can manually adjust the G92 line in your GCode file and then print the file. This will allow you to adjust the Z height without re-slicing the file. Make small adjustments, print the file, and then try again. Once you have the correct G92 settings, open your start.gcode file which is located at: replicatorg-0023/skein_engines/skeinforge-35/skeinforge_application/prefs/YOUR-PROFILE-NAME/alterations/start.gcode

Remember: once you change the start.gcode, you will need to regenerate any GCode for models that you want to print using the Raftless technique.

Step 7: Print and Trim

One of the only downsides to Raftless printing is that the first few layers bulge out slightly. This is because the first layer is slowed down, and Skeinforge has no support for automatically moving the first layer in to compensate. Hopefully it will be fixed in some future version, but for now it is very simple to trim the first layer with an exacto knife to get the proper size.

Thingomatic Firmware Installation

Save yourself some time

Before the electronics are installed inside the base of the Thingomatic, first upload the latest version of the firmware to your boards. This step will also serve as an electronics power-on test.

Get the ReplicatorG Software

You'll be using the built-in firmware uploader in ReplicatorG to update the firmware on the boards. Once you have ReplicatorG installed, can upgrade to the latest version.

New! Install drivers for Arduino Mega 2560

If you're received a new Arduino Mega2560, the drivers unfortunately do not install automatically the way they usually did for the Arduino Mega 1280.

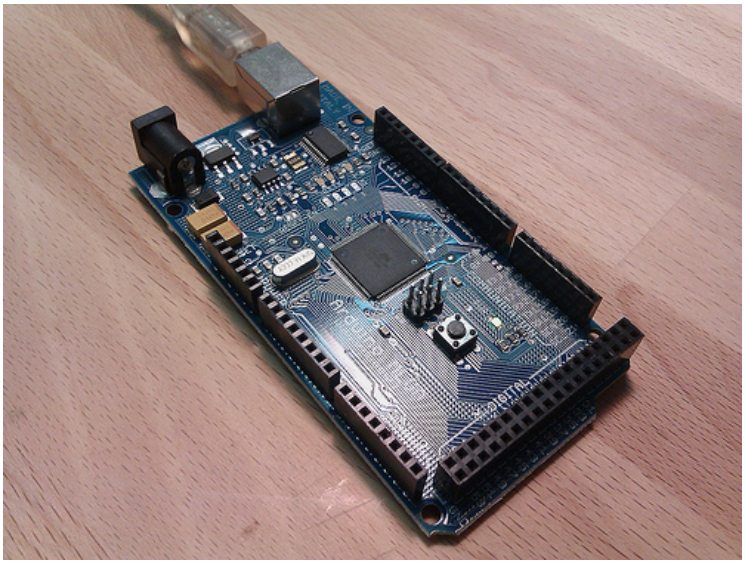
Take a quick look at your board — it will say "2560" on both sides, and the back has some cool-looking white decoration on it. If that's what you've got, don't fear — you've already downloaded the drivers with ReplicatorG. They're found in the "drivers" folder within the distribution.

Install these like any other driver. On Windows, you can follow the prompts the first time you plug in the Arduino board, or go to your device manager and find the Arduino there (look for a big yellow exclamation point.) Installation will be a bit different on Mac, but you should still have the drivers waiting for you in your ReplicatorG download. We trust you Linux types to figure this out on your own.

Install the Firmware on the MakerBot Motherboard

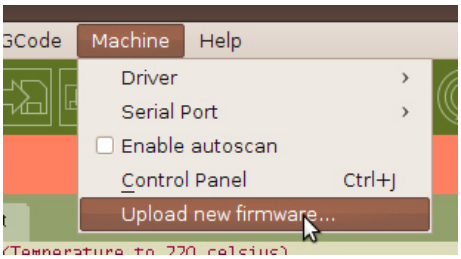
Prepare the MakerBot Motherboard

Disconnect any other USB serial devices your computer's USB ports. Next, attach the Thingomatic Arduino MEGA to your computer with the USB cable. If you have already created the MakerBot Motherboard v2.4 Stack which locks the Arduino MEGA to the bottom of the MakerBot Motherboard PCB, then you will use the MakerBot Motherboard v2.4 reset button instead of the one on the Arduino MEGA when the directions call for this action. Otherwise the directions for both route are identical.



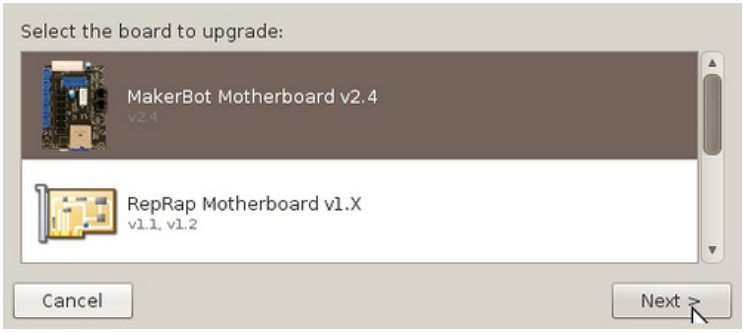
Installing the firmware

Start ReplicatorG and Select "Machines > Update Firmware..." from the menu.



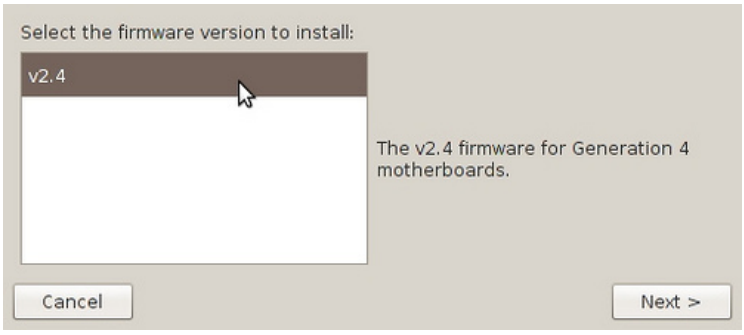
Select the board and version you're updating.

Please note: that there is a new, separate set of firmware for the Gen4 Motherboard with Arduino 2560.



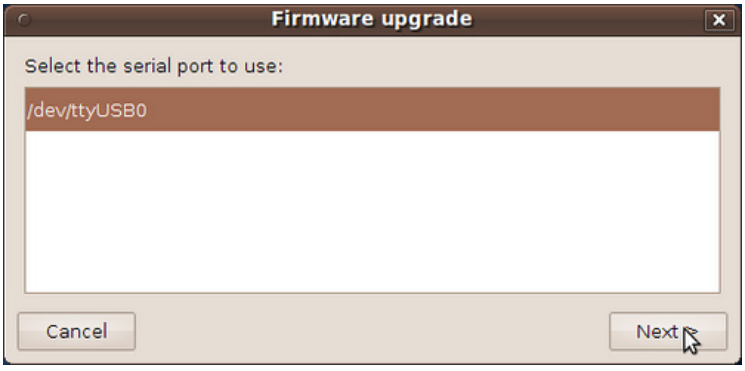
Select the version of the firmware you'd like to upload.

This should ordinarily be the one with the highest number, which is the latest version.



Select the serial port that represents your Motherboard.

Your Motherboard will appear as a serial port on your computer. Usually this will be the last one listed. The names will vary from platform to platform.



Upload The Firmware

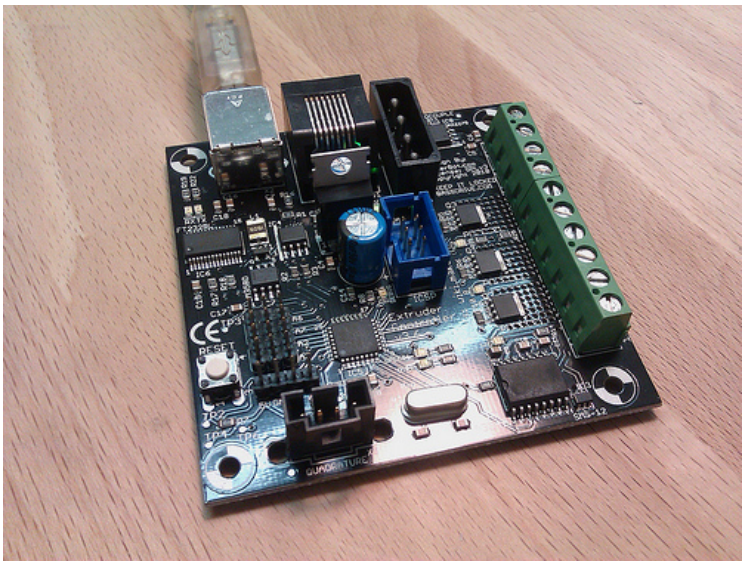
This is the tricky step. Try to press the reset button at the same time as you click the upload button.

Be patient; it may take a minute or two to upload the firmware. If the upload succeeded, you should see a message indicating success. If you see a message indicating failure, try again! Getting the timing of the reset is important; it may take a few tries to get right.

Install the Firmware on the Extruder Controller

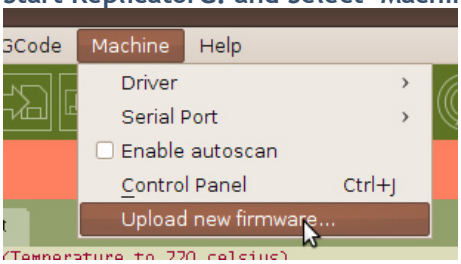
Prepare the Extruder Board

Disconnect your MakerBot Motherboard from your computer's USB port. Now, attach the Extruder Controller v3.6 to your computer with the USB cable. You're ready to go!

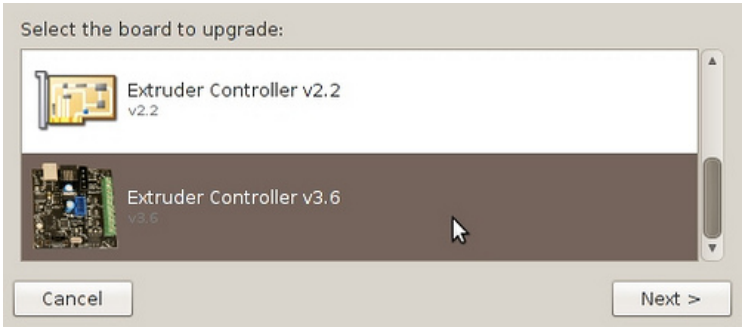


Installing the firmware

Start ReplicatorG. and Select "Machines > Update Firmware..." from the menu.

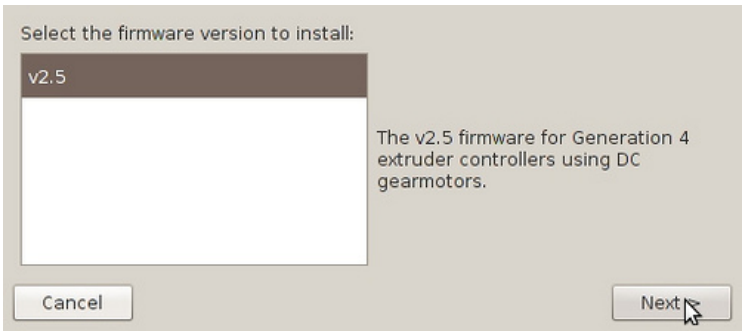


Select the board and version you're updating.



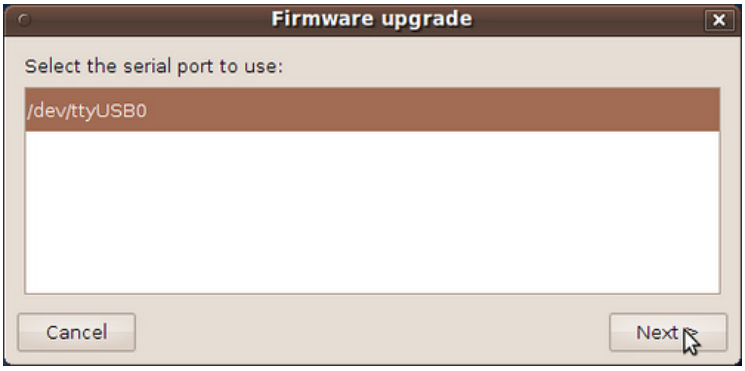
Select the version of the firmware you'd like to upload.

This should ordinarily be the one with the highest number, which is the latest version.


Select the firmware version to install:
v2.5
The v2.5 firmware for Generation 4 extruder controllers using DC gearmotors.
Cancel
Next >

Select the serial port that represents your Extruder Controller.

Your Extruder Controller will appear as a serial port on your computer. Usually this will be the last one listed. The names will vary from platform to platform.



Click on the upload button.

You should not have to manually press the reset button on the Extruder Controller.

Be patient; it may take a minute or two to upload the firmware. When you're done, you should see a message indicating success.